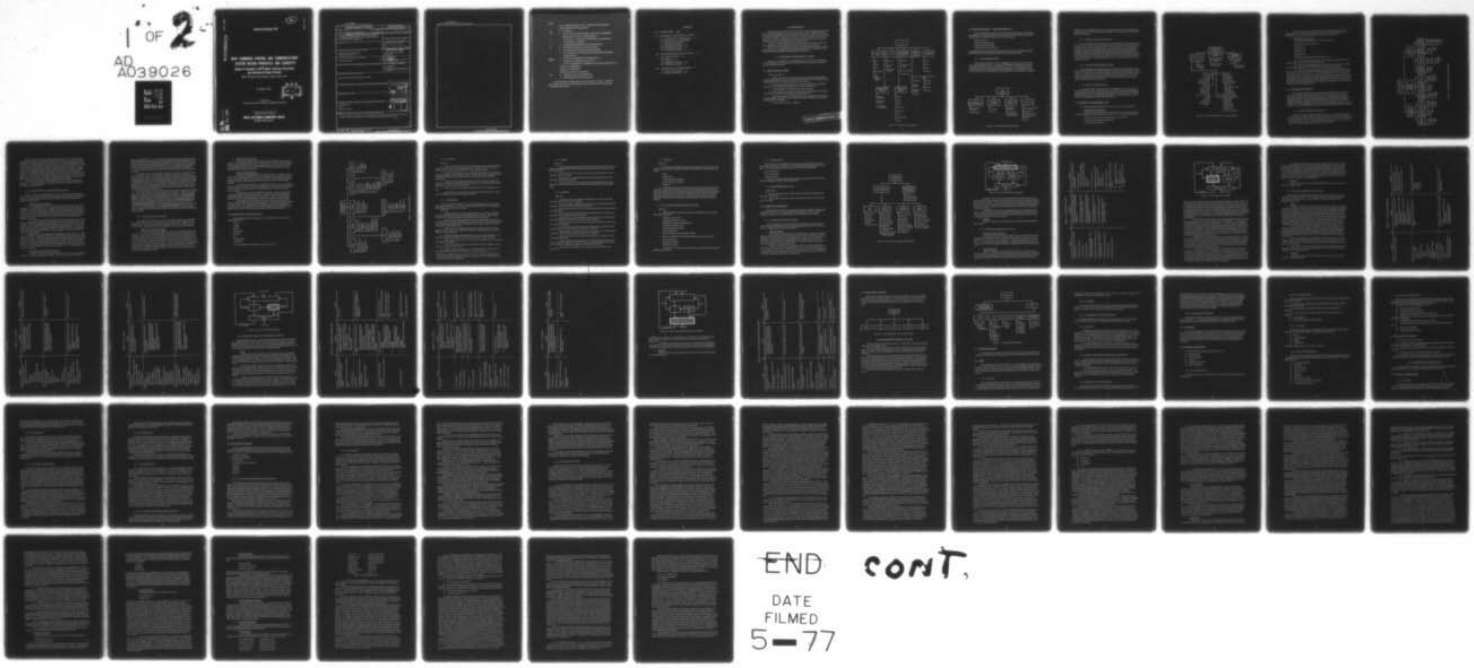
AD-A039 026 NAVAL ELECTRONICS LAB CENTER SAN DIEGO CALIF F/G 17/2
NAVY COMMAND CONTROL AND COMMUNICATIONS SYSTEM DESIGN PRINCIPLE--ETC(U)
AUG 76

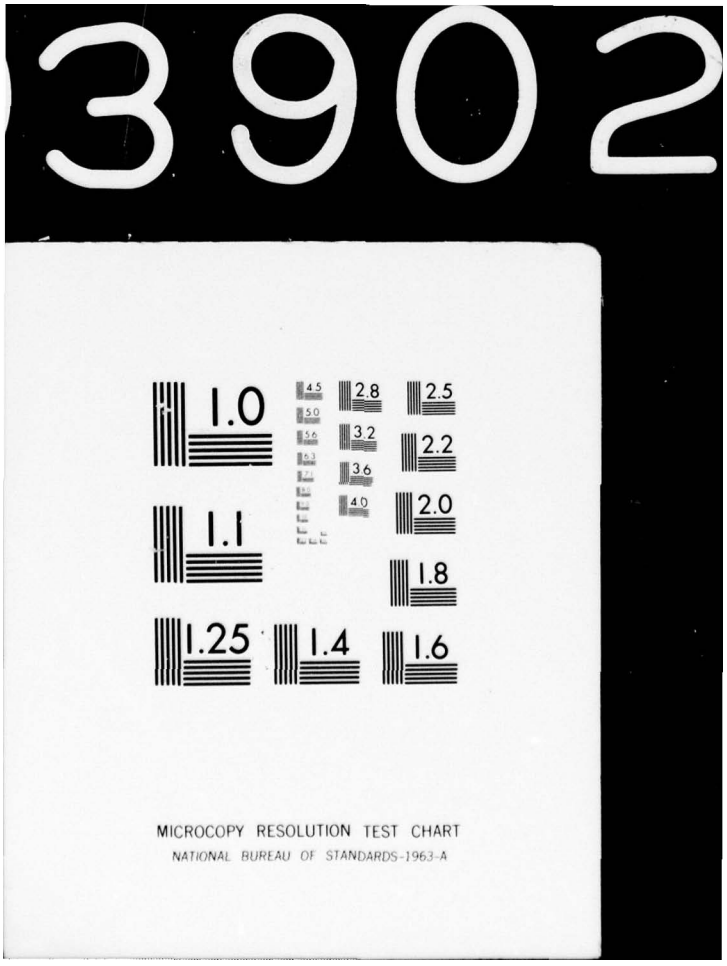
UNCLASSIFIED NELC/TD-504-VOL-4 NL

1 OF 2
AD
A039026



END CONT.

DATE
FILMED
5-77


3902
1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

NELC / TD 504

Technical Document 504

NAVY COMMAND CONTROL AND COMMUNICATIONS SYSTEM DESIGN PRINCIPLES AND CONCEPTS

Volume IV: Appendix C—NC^{3}N Nodes—Functions, Subsystems, and Architectural Design Concepts

Naval Warfare Effectiveness Group (Code 233)

15 August 1976

Prepared for

NAVAL ELECTRONIC SYSTEMS COMMAND (PME 108)

NAVAL ELECTRONICS LABORATORY CENTER
San Diego, California 92152

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
NELC Technical Document 504 (TD 504)

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

(14) NELC/TD-504-Vol-4

4. TITLE *(and Subtitle)*
(6) NAVY COMMAND CONTROL AND COMMUNICATIONS SYSTEM DESIGN PRINCIPLES AND CONCEPTS
(Volume IV, Appendix C, NC^3N Nodes – Functions, Subsystems, and Architectural Design Concepts)

5. TYPE OF REPORT & PERIOD COVERED

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)

8. CONTRACT OR GRANT NUMBER(s)
(16) F21241
(17) SF21241402

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Naval Electronics Laboratory Center
San Diego, California 92152

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
62721N, F21241, SF21241402
(NELC Q239)

11. CONTROLLING OFFICE NAME AND ADDRESS
(11)
NAVELEX (PME 108)

12. REPORT DATE
15 August 1976

13. NUMBER OF PAGES
60 (12) 60 p.

14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)*

15. SECURITY CLASS. *(of this report)*
UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*
VS A038 968
Approved for public release; distribution is unlimited
403 940

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

ACCESSION for
NTIS White Section ☑
DDC Buff Section ☐
UNANNOUNCED ☐
JUSTIFICATION
BY
DISTRIBUTION/AVAILABILITY CODES
Dist. AVAIL. and/or SPECIAL
A

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*
Command control
Navy C3
Telecommunications

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*
This is volume IV of an eight-volume document on Navy C^3 concepts. Volume IV discusses the nodes, their functional breakdown, and the organization of the subsystems to accomplish the functions.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

Volume	
I	NAVY COMMAND CONTROL AND COMMUNICATIONS SYSTEM DESIGN PRINCIPLES AND CONCEPTS
II	A. GLOSSARY
III	B. EVALUATING AND SELECTING THE MIX OF TRANSMISSION MEDIA FOR THE NC^3N–A METHODOLOGY
IV	C. NC^3N NODES – FUNCTIONS, SUBSYSTEMS, AND ARCHITECTURAL DESIGN CONCEPTS
V	D. AUTOMATED ORDERWIRE CONCEPTS FOR NC^3N
VI	E. NETWORKING PRINCIPLES AND FEATURES
	F. DATA BASE MANAGEMENT CONSIDERATIONS
	G. NC^3N USER DATA AND INFORMATION EXCHANGE NETWORK REQUIREMENTS
VII (Secret)	H. NUCLEAR ENVIRONMENT CONSTRAINTS (U)
	I. THREAT ENVIRONMENT FOR NC^3N (U)
	J. NC^3N VULNERABILITY TO JAMMING, DECEPTION, INTERCEPT (U)
	ANNEX A OF APPENDIX B
	ANNEX E OF APPENDIX E
	ANNEXES C AND D OF APPENDIX G
VIII	K. ARCHITECTURAL ALTERNATIVES
	L. DESCRIPTIONS OF R&D INITIATIVES

This is volume IV of an eight-volume document on Navy C^3 concepts. Volume IV discusses the nodes, their functional breakdown, and the organization of the subsystems to accomplish the functions.

CONTENTS

C1.0 INTRODUCTION

This appendix describes the generic NC^3N nodes in terms of functions (what is done in the node) and subsystems (what is constructed to make a node). A node can be defined and/or described at any level in a network hierarchy. Therefore, the generic node functional or subsystem description could represent any node anywhere in the hierarchy insofar as it is a generic and exhaustive shopping list of functions and subsystems from which selections can be made based on the requirements of that particular node.

In the case of this document the focus is on the platform-level node; ie, all the electronics installed on a platform whether ship, aircraft, surface craft, or manpack.

Section C2.0 of this appendix is the functional breakdown and description, section C3.0 is the subsystem breakdown and description, and section C4.0 is the discussion of an architectural concept leading to distributed network architecture.

C2.0 FUNCTIONAL STRUCTURE OF A C^3 NODE

Figure C2-1 is the first-level functional breakdown of the C^3 node, showing seven first-level functions. The following sections describe these seven functions in some detail in terms of their subfunctions.

C2.1 EM AND ACOUSTIC LINKS

The link functions are:

a. Radio and acoustic communication links for exchanging messages and data among cooperating nodes. Uses of such links include navigation to the extent that Δt from known locations can be determined on the basis of time markers.

b. Radio detection and ranging for the purpose of navigation, locating objects, and, in some cases, identifying them on the basis of return signal analysis.

c. Sound navigation and ranging for same purpose as radar.

d. Passive ESM for the purpose of passively intercepting enemy (or other) radiations for ELINT, SIGINT, tactical targeting, and identification, etc.

e. ECM for the purpose of interfering with an enemy surveillance or command control capability or operation or for the purpose of ELINT or SIGINT by observation of cause-and-effect relationships.

f. Acoustic communication – same as (a).

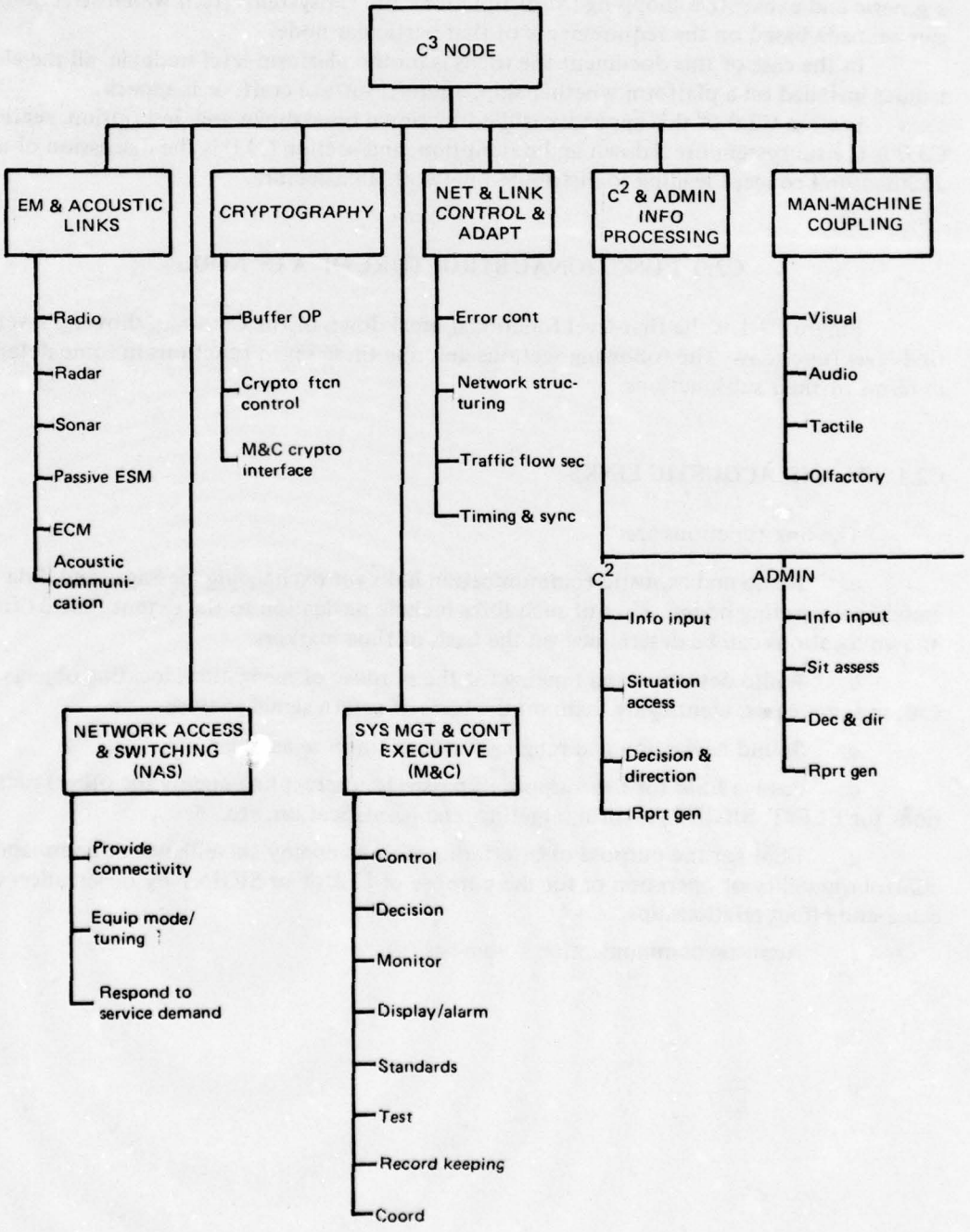


Figure C2-1. C^3 nodal functions and processes.

C2.2 NETWORK CONTROL – LINK ADAPTATION (LA)

LA is a programmable operation in the node in which data streams are conditioned and managed for rf transmission and baseband reception. These conditioning and management functions are outlined in figure C2-2 and are the following:

Link error control

Timing and synchronization

On-line network structuring

Traffic flow (or activity) security

Link adaptation functions performed within the node are to some extent similar to those performed by a front-end processor in systems such as the ARPA network.

C2.2.1 LINK ERROR CONTROL

Under direction of M&C, LA encodes outgoing data for error correction and detection at the receiving destination. Conversely, LA will decode incoming error control patterns and provide error correction and detection. These error control techniques will include ARQ, block and convolutional coding, parallel diversity, or combinations of these. Various detection/correction algorithms will be employed, such as majority vote, threshold

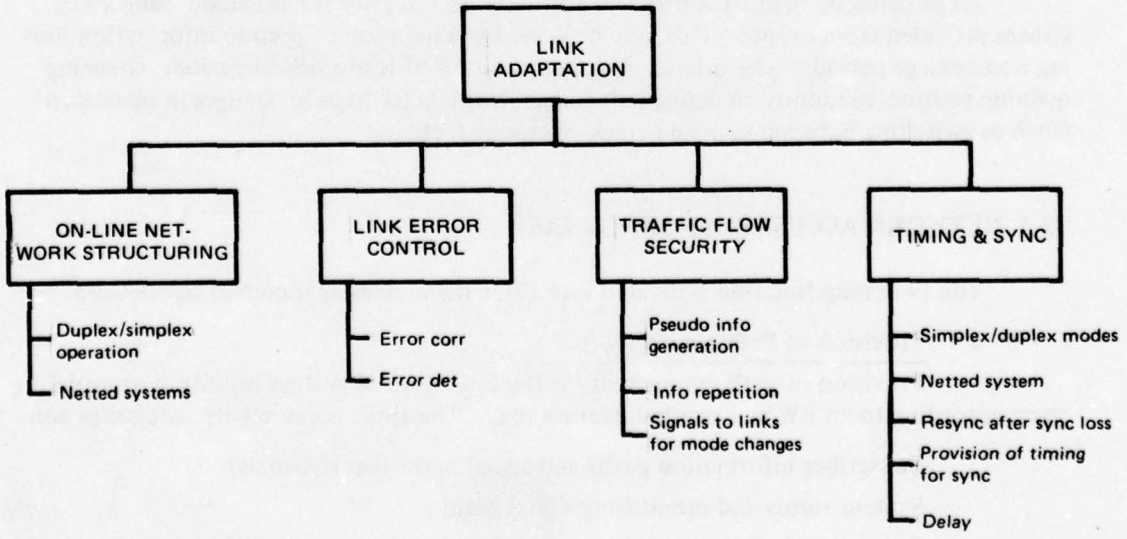


Figure C2-2. Link adaptation functional breakdown.

decoding, and error trapping techniques. The selection of the type of error correction and detection scheme will depend on network structuring, channel characteristics, and allowable delay for the message.

C2.2.2 TIMING AND SYNCHRONIZATION

The LA function derives block and frame synchronization for data transfer throughout the node. For example, the link function derives frequency, phase, and bit timing necessary for demodulation on the incoming data if the system is synchronous. The LA must then accept the data and derive information such as start of block and frame synchronization necessary for data transfer throughout the node. Path delay (Δt) determination for near-real-time synchronous transmission if not available from M&C will be determined by LA for network operation and time division transmission. (Note: any Δt derived by any subsystem is transferred to M&C for correlation with information from other sources and for use elsewhere in the node.)

C2.2.3 ON-LINE NETWORK STRUCTURING

LA provides logic for sequencing messages in networks (ie, polled network) and provides routine control and data management once connectivity has been established. In other words, once the node has been properly configured for participation in a network, routine control is transferred to LA for network bookkeeping and data management. The degree of this on-line structuring will depend on the type of operation and the various requirements for keeping the network functioning.

C2.2.4 TRAFFIC FLOW SECURITY

As directed by M&C, LA provides for network integrity for the node using a key stream provided from crypto. This may be done by generation of pseudo information during nonmessage periods. The interleaving, permutation of transmission periods, changing queuing routine, randomly changing radar signal formats to disguise changes in operation (such as switching between scan and track operation), etc.

C2.3 NETWORK ACCESS SWITCHING (NAS)

The switching function is divided into three main areas as shown in figure C2-3.

a. Provision of Path connectivity

Provision of path connectivity is the operation of system initiation or initial setup according to an EW and communication plan. The three connectivity categories are:

Subscriber information paths and signal paths (eg, rf signals)

System status and monitoring signal paths

System control signal paths

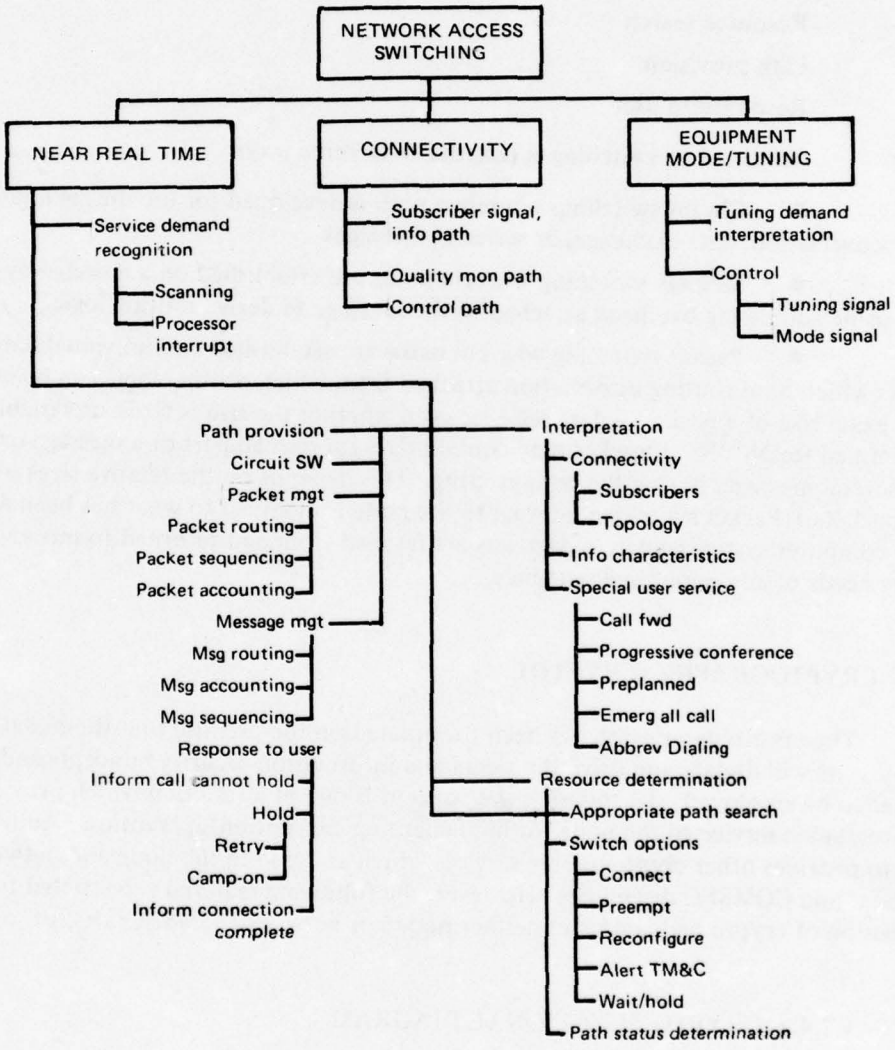


Figure C2-3. Functional breakdown for network access switching.

These functions are essentially non-real-time and semipermanent and are typically changed only when a new EW or communication plan or modification thereof is introduced or a failure causes M&C to direct the setup of an alternate path.

b. Operational Time Switching

Operational time switching has five functional categories:

Recognition of demand for service

Interpretation

Resource search

Path provision

Response to user

This kind of switching is referred to in three ways:

- Circuit switching wherein a path is developed for use during a relatively long conversation, data exchange, or series of messages
- Message switching wherein paths are established on a message-by-message basis using addressing overhead attached to the message to derive routing logic.
- Packet switching wherein paths are established for individual data sets or blocks which have routing information attached from which routing logic can be obtained. (The exact size of a packet – data set – or even whether the size is fixed or variable is not yet defined for NC^3N. A packet may contain data for a small part of a message or may contain several messages having the same routing. This depends on the relative sizes of messages and packets.) Packet switching internal to the node is identical to what has been done in large computer systems wherein data sets are fetched from and returned to buffer according to the needs of an operational sequence.

C2.4 CRYPTOGRAPHY (CRYPTO)

The crypto description has been formulated on the premise that the nodal C^3 requirements will dictate and drive the signal and information security concepts and techniques to be employed. To this end, the concept is one of a function which provides a cryptographic service to the node for implementing encryption/decryption. Additionally, crypto provides other cryptographic services which are used in the node and network SIGSEC and COMSEC disciplines. However, the following material is restricted to a general discussion of crypto and addresses neither node nor network COMSEC/SIGSEC operation.

C2.4.1 CRYPTO FUNCTIONAL DIAGRAM

Figure C2-4 is the functional organization of crypto. All modes of operation are controlled by M&C. That is, crypto executes subroutines which provide the cryptographic service demanded by M&C, but does not direcly engage in any decision processes affecting the nodal operations. However, the performance of cryptologic processes, such as cryptovariable and keystream generation, are the sole province of crypto.

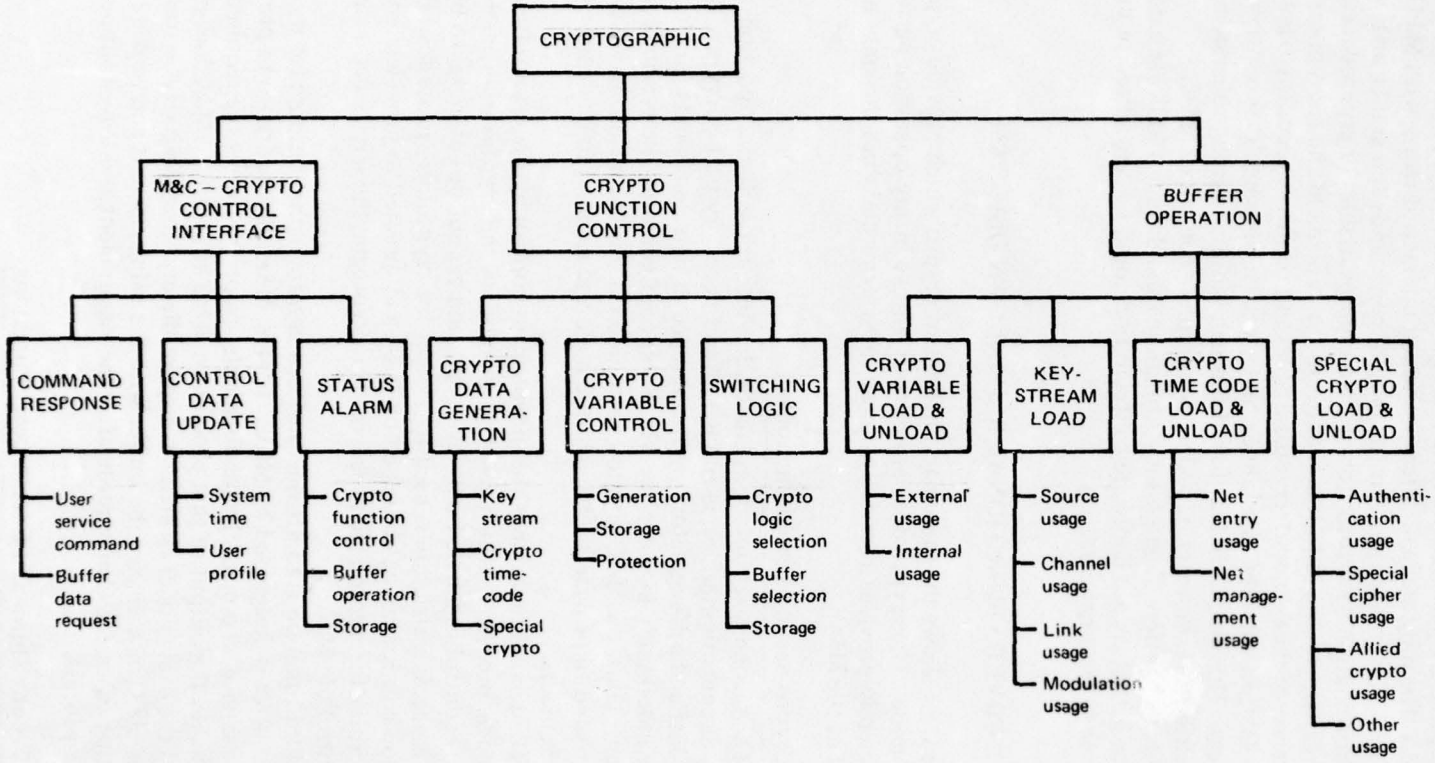


Figure C2-4. Cryptographic functional breakdown diagram.

Interaction between the various functional elements of crypto is supervised by the crypto function control. This interaction consists of data, commands, and status and must be handled via properly partitioned sections of the system. (Such partitioning is either logical or physical.) The only element of crypto which interacts directly with M&C is the crypto function control. This interaction consists only of commands and status, and, as a special case, control information to crypto storage. No cryptovariables or cryptodata are supplied to M&C. The cryptographic output is keystream (to implement the encryption and decryption functions of the C^3 node), cryptovariables and crypto time codes (to implement node and network COMSEC/SIGSEC discipline), and special cryptodata (for nonstandard cryptographic services). This output is distributed through buffers to using subsystems within the node. The distribution, however, is under the complete control of M&C.

The functional flow diagram shows functions and does not indicate hardware configuration. The actual crypto hardware configuration could be centralized or distributed depending on node requirements.

C2.4.2 CRYPTO FUNCTIONAL BREAKDOWN DIAGRAM

Figure C2-4 shows the principal functions of crypto which provide cryptographic service to the node. The crypto does not directly engage in any operation performed within the node. It merely provides a service to those subsystems and does this only under the direct control of the M&C.

a. Crypto Second-Level Functions

The second-level functions are divided into three distinct portions. The first, the M&C-crypto control interface, is an integral and necessary part of crypto. While crypto is self-operational and self-regulatory, in the absence of new commands (or altered status) it will operate indefinitely in the mode which was configured by the previous command and status situation. Thus, the M&C-crypto control interface is the means by which the crypto function is directed to provide a service to the node and respond to changing COMSEC/SIGSEC requirements.

The second function is the independent control by the crypto function control of the cryptologic processes (which generate cryptodata and cryptovariables) which service the node. This control is handled only by crypto and cannot be entered into by any other function. This access restriction is used to protect the cryptologic processes. However, the crypto function control can be addressed by M&C and directed to provide a cryptographic service (the result of a cryptologic process). The cryptographic service is then distributed under the control of M&C.

The third function is to provide buffered cryptographic outputs for use in the node. The user of a buffer is designated by M&C. That is, M&C directs crypto to provide a certain cryptographic service. To meet the requirements, crypto performs the required cryptologic process and makes the output of that process available at a buffer. M&C has, in the meantime, directed the using nodal element to acquire the cryptographic service from the buffer. In some cases, M&C may engage the buffer with the using element and provide continuous control; in other cases, M&C may retire after the engagement and leave routine buffer control to the routine process.

b. Crypto Third- and Fourth-Level Functions

The M&C crypto control interface function has three parts. First, crypto must respond to two classes of commands from M&C. User service commands direct crypto to

provide a cryptographic service to a specified user, and buffer data requests engage the interface between a buffer and a using element. Second, M&C supplies important control information to update system time and the user profile stored in a memory while under control of crypto. The user profile uniquely identifies the user of cryptographic services and lists technical parameters affecting the nature of the service when provided. Third, M&C receives status alarms which indicate abnormal functioning of the crypto functional control or storage.

The crypto function control has three functions: generation of cryptodata, control of cryptovariables, and control of switching. Cryptodata are classed as being either keystream, cryptotime codes, or special crypto outputs (used in authentication and nonstandard cryptographic processes). The control of cryptovariables involves their generation, storage, and cryptographic protection. The switching is directed to select the cryptologic to be used in providing the demanded service and connects the cryptologic to the buffers which will provide the service to the requiring nodal element. Control over storage of data used in the cryptologic processes is also included. All these functions are under the sole and exclusive jurisdiction of the crypto function control.

The buffer operation function of crypto involves the control, loading, and unloading of the buffers which provide the cryptographic service to the node. The buffer operation function has cognizance over the following categories of buffering: (1) the cryptovariable buffer, which accepts and delivers cryptovariables to and from users external to the node and delivers cryptovariables to users internal to the node; (2) the keystream buffer, which delivers keystream for encryption, decryption, link protection, or control of modulation; (3) the crypto time code buffer, which provides time codes which can be used to enter cryptonets and manage cryptonets; and (4) the special crypto buffer, which allows the node to have the special cryptographic services associated with authentication, special cipher systems, communication with allies employing various cryptologics, and other limited-usage special functions.

C2.4.3 CRYPTO INTERFACE MATRIX

The only control provided to the crypto is from M&C. Likewise, crypto provides status only to M&C. However, information can be exchanged between crypto and some node functions. This information exchange is unilateral with respect to M&C and links, bilateral (with exceptions regarding type of information) with the other functional areas.

a. M&C Control and Status Interfaces

The only control inputs from M&C are user service commands and buffer data requests. The user service commands identify a user and indicate the cryptoservice to be provided – either cryptodata or cryptovariable. The buffer data requests engage the interface between the crypto-controlled buffer and the using functional area of the node. This control allows M&C to switch users while keeping cryptoservice current in the buffers. The status output from crypto M&C is alarm status: control overload (such as being swamped by user service commands), excessive demands on crypto-associated buffers; storage overload; cryptologic and storage malfunctions; accidental or malicious intrusion into the cryptologic processes or storage; or illegal commands (ie, user service command inconsistent with user technical requirements on file).

b. M&C Information Interfaces

The crypto function has an information input from M&C. This information consists of system time and the user profile. The user profile contains a unique user identity code and technical data important to the cryptologic processes, such as bit rate, real-time or non-real-time operation, and use of the service (link encryption, modulation control). C&M updates the user profile as required.

c. Other Information Interfaces

Keystream is used elsewhere in the node for information encryption or decryption functions associated with MMC, IP, and LA, respectively, or for modulation control (such as AJ or LPI by the links). Keystream may be made available outside the cryptologic processes so that data handling and encryption/decryption can be done with maximum flexibility.

The crypto time code gives the cryptographic time-of-day to a user. This time code is information which can be used in transmissions to allow node entry into communication nets and the control of crypto synchronization within a net. IP and LA would be users of such data.

Cryptovariables may be transmitted between nodes. These variables may be net control variables or individual user variables. The variables enter crypto through IP (under C&M control) and are processed by the cryptologic. Crypto can provide cryptovariables to the net (via IP and under M&C control) or to internal users (such as MMC).

Special crypto provides cryptoservice which cannot be provided by normal means. LA, IP, and MMC may require special crypto data. Cryptoservices provided include those which involve data manipulation by cryptographic means, such as authentication and cipher systems. Also, special keystreams may be generated for use with allied or nonstandard crypto systems, weapons systems, antispoof command control links, etc.

C2.5 MANAGEMENT AND CONTROL (M&C)

M&C provides control and management for all configuration of a node under the following subfunctions:

Control
Decision
Monitor
Display/alarm
Standards
Test
Record keeping
Coordination

A functional breakdown of the M&C is shown in figure C2-5.

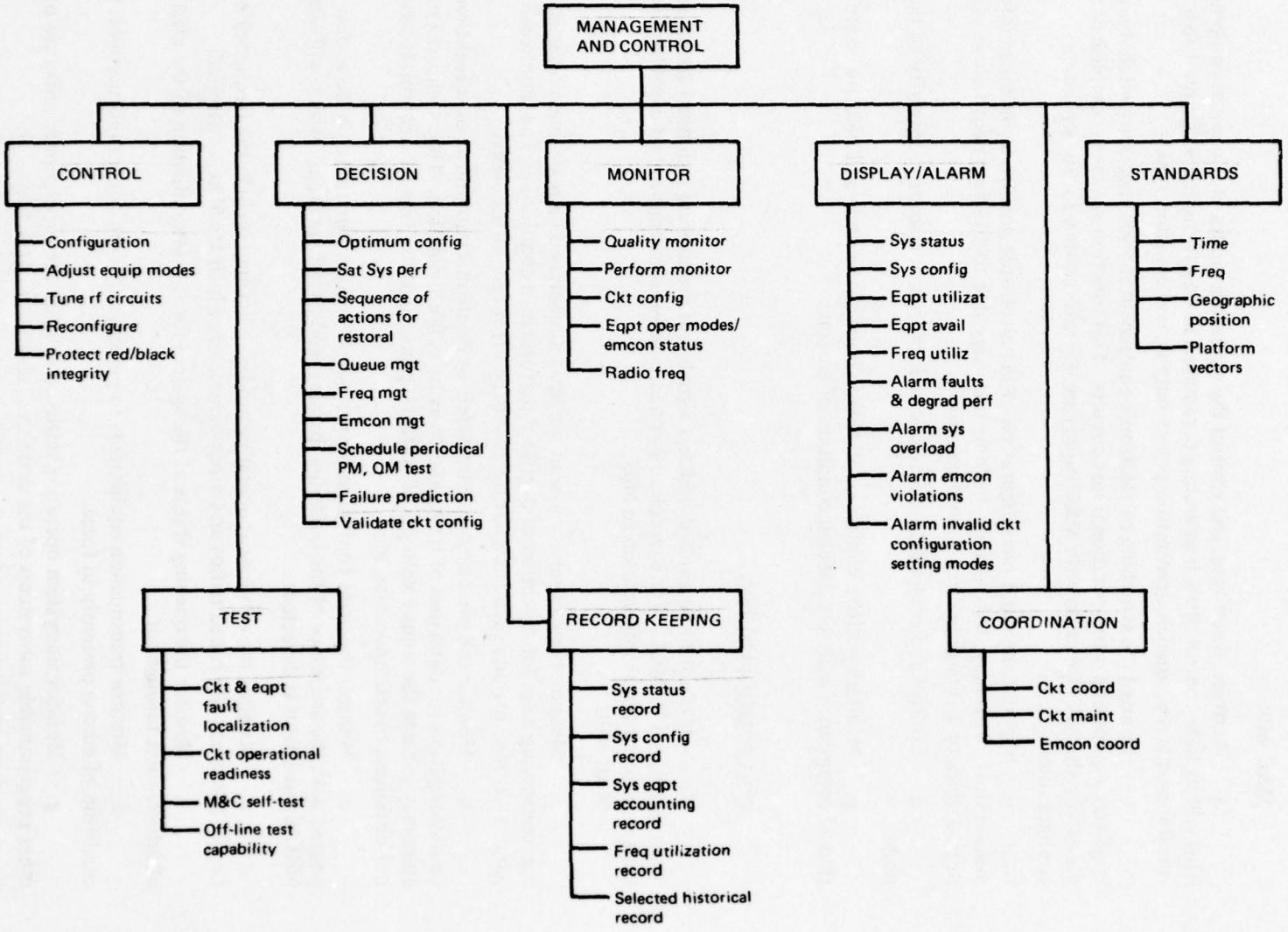


Figure C2-5. M&C functional breakdown.

C2.5.1 CONTROL*

M&C will:

a. Activate, deactivate, and control the operating modes of the nodal equipment. This will include control of such operating parameters as on-off, tuning a rf circuit to a specific frequency, squelch, transmitter power output, and emission type.

b. Control the switching of electronic equipment interconnection switchboards to configure equipment into operational test circuits. This function will be accomplished completely by the NAS function with M&C supplying the information for the optimum configuration.

c. Control the rapid reconfiguration of the electronic posture to optimize information flow. This may, for example, become necessary due to an equipment failure, signal degradation, or a change in circuit requirements.

d. Control and protect the electronic and information security integrity of the node.

e. Maintain positive control of all rf emissions of the node and indicate confirmation of compliance with any defined condition of emcon.

C2.5.2 MONITORING

Note: It is not to be implied that test signals and monitoring functions are necessarily originated by M&C. For example, equipments possessing built-in test capability will send monitored status information to M&C.

M&C will:

a. Measure basic in-service circuit quality parameters on a real-time, noninterfering, continuing basis for detection of degraded performance conditions. The monitored parameters may include such things as the baseband error rate of the signal.

b. Monitor the performance parameters of electronic equipment on a real-time, continuing basis for detection of faults and to assist in fault isolation. The monitored parameters may include the supply voltages, rf voltages, phase lock, transient detection, blower motor failure, overtemperature, etc.

c. Monitor the node configuration periodically. This function may be accomplished with the assistance of the buffering, bussing, and switching group, which will alarm M&C when a fault is detected.

d. Monitor the equipment operating modes. This may include the frequency setting of automatically tuned radios and other parameters which are M&C controlled.

e. Monitor the queuing status of the system for the determination of the efficiency of information throughput.

f. Monitor transmission equipments to periodically confirm compliance with the condition of emcon presently in force.

g. Monitor subsystem operating status to validate the programmed software of other programmable subsystems of the node to ensure their correctness.

---

*Control in this functional area is control by exception. It is limited to system initialization and interventions to restore or change routine procedures when a system aberration occurs or a change in the EW or communication plan is to be implemented.

C2.5.3 TESTING

M&C will:

a. Test circuit and equipment electrical characteristics both on and off line, using built-in test equipment (BITE) where possible to localize faults and isolate them to the replaceable item level.

b. Test each circuit after initial setup to assure operational readiness before turning the circuit over to the user.

c. Provide a self-test capability which will exercise a substantial portion of the subsystem to assure proper operation.

d. Provide an off-line test, measurement, and analysis capability to be used by C&M operator manually when the system is operating in a degraded mode without failure alarm.

C2.5.4 DECISION

M&C will:

a. Determine the optimum equipment configuration by analysis of given user circuit requirements and resource availability.

b. Have the capability to determine the satisfactory system circuit and equipment performance levels, fault, and out-of-tolerance conditions.

c. Provide failure prediction for circuits and equipment based on trend analysis of past measurements.

d. Determine the actions necessary for graceful degradation of the node due to equipment faults or degraded operating conditions.

e. Determine the sequence of actions necessary for the restoral of normal service after system repair, following fault or degraded operating conditions.

f. Provide queue management to control system utilization under overload conditions (demand scheduling).

g. Provide frequency management to assure proper frequency separation and to prevent frequency and intermodulation interference in rf channels.

h. Provide high-level management of electronic defensive measures including levels of protection against detection (emcon), interference, and intrusion.

i. Schedule performance monitoring, testing, maintenance, and other necessary scheduled control actions.

j. Validate equipment parameters and circuit configuration prior to implementation, to assure proper equipment compatibility with respect to operating frequency, bandwidth, modulation, impedance matching, line voltage, and current levels.

k. Possess decision capability to ensure conformance to security and emcon requirements.

C2.5.5 DISPLAY

M&C will:

a. Provide system management and control personnel with all the necessary information on which to base the decisions required to operate the node. This information will include:

Status

Configuration

Circuit equipment utilization

Circuit equipment availability

Frequency

b. Alarm M&C personnel of conditions which require their immediate attention. For example, alarms may be provided for such conditions as degraded circuit performance and faults; invalid configurations, equipment settings, and modes; system overloads; attempted security violations; and any demand or scheduled action whose subsequent execution would violate the present node emcon status. The latter alarm would be made available prior to the operational release of equipments whose activation would cause the emcon violation.

C2.5.6 RECORD KEEPING/REPORT GENERATION

M&C will:

a. Have the capability to store and maintain for immediate retrieval the following real-time information:

Equipment status record

Configuration and status record

Equipment inventory accounting record

Frequency utilization record

User address directory record

Traffic volume data

Circuit and link activation/deactivation status

b. Maintain a historical record of selected information which may include:

System performance

Frequency utilization

Maintenance actions

The time frame of coverage for these records is to be determined in the system design phase of this task.

C2.5.7 COORDINATION

M&C will provide the capability for composition and interpretation of secure non-message character strings to be transmitted to or received from C&M configurations of other nodes. These character strings would contain information relating to internode:

Circuit coordination

Circuit maintenance

Node emcon status

Current node capability in terms of any recent physical status change which would enhance or degrade its last reported mission capability.

C2.5.8 STANDARDS MAINTENANCE

The M&C will:

a. Maintain or provide central timing, frequency, location, and platform vector information for the node.

b. Provide other node functions with time, frequency, location, and vector information as required.

C2.6 INFORMATION PROCESSING

Information processing can be conveniently divided into two major functional areas as illustrated in figure C2-6 – tactical/strategic and administrative. Only the tactical/strategic information processing functions will be further analyzed in this section.

C2.6.1 INFORMATION INPUT FUNCTION

As the initiating function of the C^2 process, information input comprises all procedures necessary to the preparation of incoming data and information for its eventual use in the succeeding functions as indicated in figure C2-7.

a. Scope of Function

This function encompasses the receipt, categorization, formatting, correlation, storage, retrieval, and presentation of data and information used in a node in the decision-making, decision-implementing, and monitoring process. In this context, data are pieces of uncorrelated and unrelated raw information, generally unusable in their received form; information is processed data and is usable directly. Correlation consists of collation, synthesis, and integration of data and information.

The data and information may already be on file in a node, they may be received unsolicited from internal or external sources, or they may be received in response to a query originated by any node. Data and information may be in any form compatible with exchange between humans, between machines or between humans and machines; they may be received continuously, periodically, or sporadically.

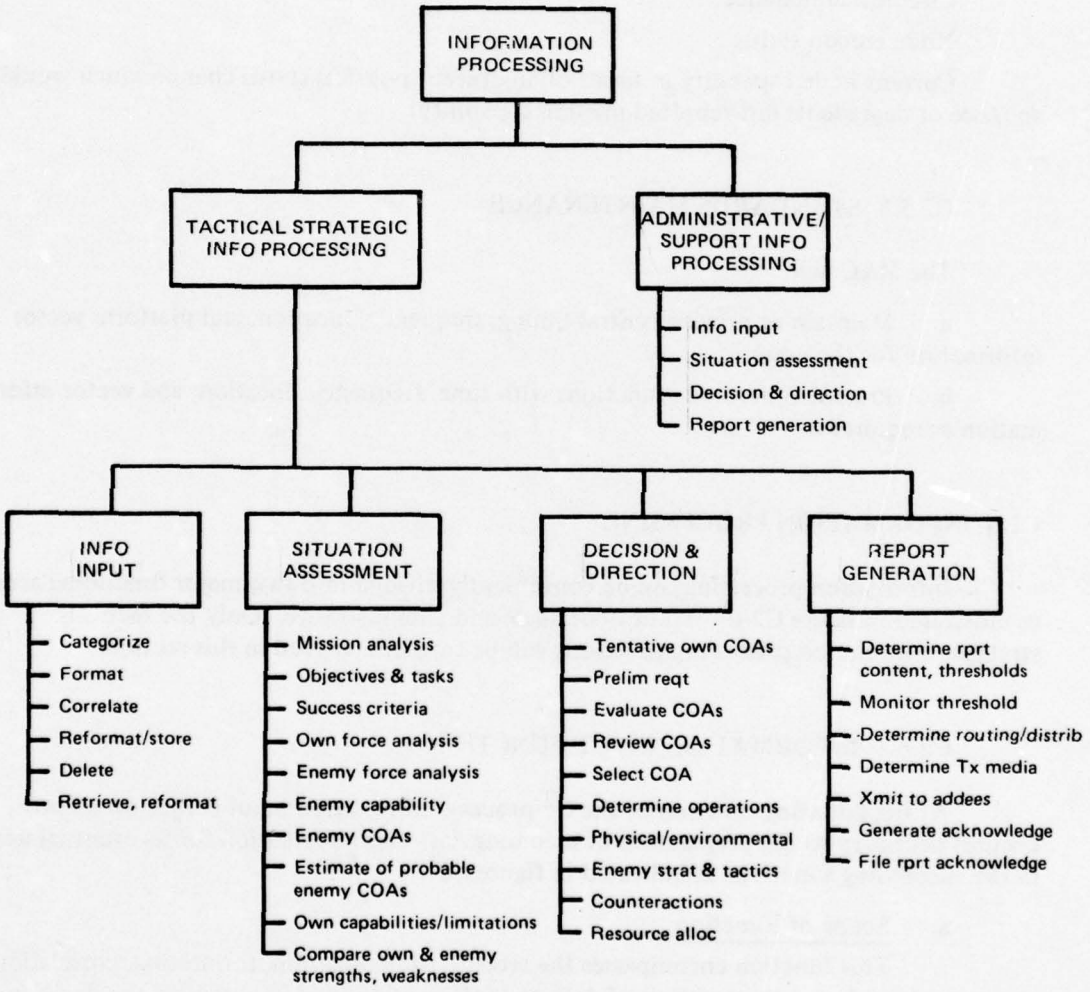


Figure C2-6. Information processing functional breakdown.

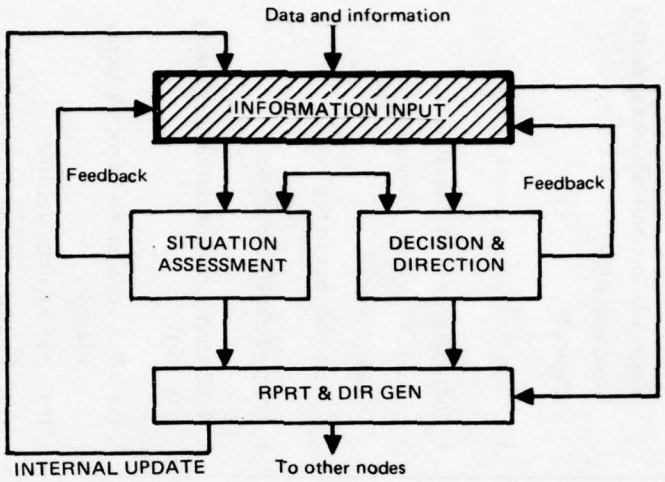


Figure C2-7. Information input function flow diagram.

Information from the information input functions is used in response to demands from the situation assessment function, the decision and direction function, and the report generation function. Outputs from these functions, in turn, may be used for internal update of the node's data base or to identify a need for data or information not then available within the node. The information may also be presented directly to a commander if the situation so dictates.

The information input function's primary purpose is to support operational aspects of command. Processing assets accorded this function may be used, however, for nonoperational purposes when such use will not degrade operational performance criteria or otherwise interfere with its primary purpose.

b. Summary

Table C2-1 summarizes the inputs, outputs, and processes of the information input function.

C2.6.2 SITUATION ASSESSMENT FUNCTION

a. Situation Assessment Function

The situation assessment function provides for the correlation of information for purposes of establishing a context for the eventual formulation of tentative courses of action. This estimate is in effect a correlation of a commander's perception of threat, his resources, and his assigned task performed either as a prelude to initiating an operation or as a response to situational changes occurring in an ongoing operation as indicated in figure C2-8.

b. Scope of Function

High-level contingency plans are formulated for various critical geopolitical ocean areas where it is clear the US must maintain sea dominance in the interest of national security and for furthering political and economic objectives. These national objectives are developed by National Command Authorities, and the operations plans, based on these

TABLE C2-1. INFORMATION INPUT.

Inputs	Processes	Outputs
1. Current own force mission, task statement, supporting material	1. Categorize received data as necessary according to type, content, source, format and handling requirements	1. Own force mission, task statement, and supporting material
2. Current force characteristics	2. Perform data formatting as necessary to allow conversion to information; convert	2. Own force characteristics Platform characteristics/capabilities Reconnaissance/surveillance capabilities Readiness state Disposition
3. Current force status	3. Correlate information with that currently residing in the data base; perform redundancy and perishability check	3. C^3 structure; own & enemy
4. Current tactical picture	4. Reformat nonredundant information as necessary and store in the data base	4. Characteristics of the operating area Political, military, economic Environmental Time/distance factors
5. Intelligence estimates, summaries, reports, updates, classified publications	5. Delete perishable information as appropriate	5. Enemy/force characteristics
6. Naval Warfare publications, books, messages	6. Retrieve information from the data base; reformat as necessary for display	6. Intelligence summaries estimate of enemy intentions
7. Operating area environmental/geographical maps and publications		7. (a) Previously employed own & enemy strategies (b) Naval warfare doctrine (c) Strategic/tactical alerts, warning indicators
8. Historical enemy strategy and tactics file		
9. Strategic/tactical alerts		
10. Current mission progress reports		

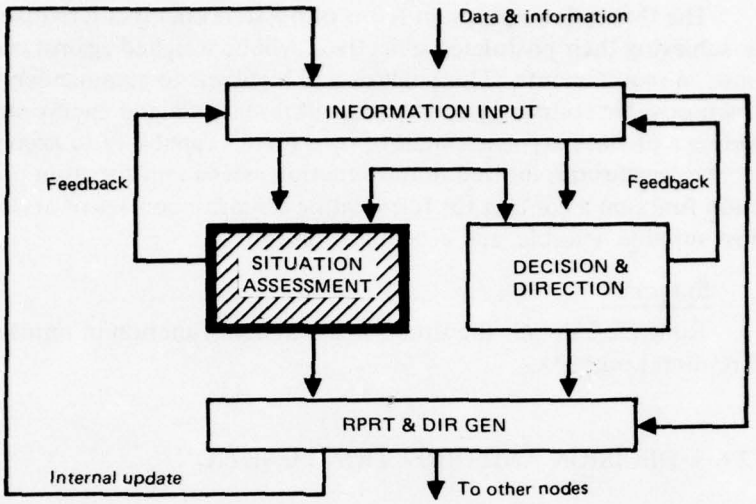


Figure C2-8. Situation assessment flow diagram.

objectives, are formulated at various levels of the military command structure. Within Navy command control, operations plans developed at the FLTCINC level cover their ocean areas of responsibility and describe the command relationships of subordinate commanders who are charged with carrying out the plans and in turn prepare lower-level and more detailed plans and operation orders for carrying out required missions and tasks. At the task element or platform nodal level, these may be no more than promulgating the platform's requirement for establishing certain communications circuits, station assignments in various dispositions, and task assignments, and schedules for such evolutions as plane guard duty and shore bombardment.

The FLTCINCs must have up-to-date detailed knowledge of all potential enemy orders of battle and tactics relevant to their ocean area responsibility. Through the provision of additional essential elements of information – ie, own fleet resources; capabilities, including the existing and projected command control and communications capabilities of own and enemy command structure; and characteristics of operating areas within the ocean-wide area of responsibilities – the FLTCINC, by making major assumptions, continually formulates estimates of the situation for his ocean area of interest. This planning is an essential element in assessing the general threat and in detecting trends which may necessitate a change in the planning or execution process.

Numbered Fleet Commanders and TF/TG/TU Commanders require a subset of the FLTCINC's information base concerning enemy orders of battle and tactics to maintain a current overall picture with emphasis on specifics pertinent to the immediate operating area environment. This observation extends to individual platforms, where knowledge of enemy intentions and capabilities is vital in the event that commanders are required to act in the absence of direction from higher authority.

Commanders must periodically assess their available resources, establish mission objectives, and evaluate ways of coping with the expected tactical situation. The FLTCINC in coordination with the responsible at-sea commanders will ascertain the current status of forces and determine whether assets of the force as constituted are adequate for carrying out assigned missions. With missions and mission guidelines established, the FLTCINC and at-sea commanders compile all related mission requirements including the threat to naval forces and the at-sea defensive posture and capabilities.

The threat description, in terms of possible enemy alternative courses of action (COA) for achieving their postulated objectives, will be weighed against own force capabilities and mission requirements. This analysis will highlight to commanders their limitations affecting own possible courses of action and will reveal own and enemy strength and weakness, providing a preliminary assessment of own force's capability to accomplish the mission objective. The conclusions derived in the situation assessment function provide the decision and direction function a context for formulating tentative courses of action and for selecting the most suitable, feasible, and acceptable option.

c. Summary

Table C2-2 details the situation assessment function in terms of its inputs, processes, and resultant outputs.

C2.6.3 DECISION AND DIRECTION FUNCTION

Drawing upon the insight and knowledge acquired in the performance of situation assessment as indicated in figure C2-9, the commander will have at his disposal broad concepts relating to potential courses of action as well as criteria with which eventually to judge their relative merits.

a. Scope

The decision and direction function is the embodiment of all processes required to formulate, evaluate, and plan the course of action whose execution will lead to the attainment of mission objectives. It is that part of the procedure which establishes how assigned tasks are to be carried out, what forces will be committed and when and where. The primary inputs for this segment of the procedure are the outputs of the situation assessment function with additional supportive inputs as required from the information input function. In performing the decision and direction function, the need for modification of earlier estimates and assessments may become apparent. The function must be so structured that changes and modifications identified are fed back into the system so that their validity can be established. This iterative procedure is a key element of the overall system concept.

Once the need for an action is established, the next step in the process is to identify who is responsible for deciding what the action shall be to ensure that the action selected is consistent with others that may already be in process. Command authority is often determined in advance by doctrine which identifies appropriate decision makers. The command determination phase in the C^2 process is crucial to the overall effectiveness of the system.

Once the need for action is identified and the decision maker has formulated, reviewed, and selected among the alternative courses of action, it is then necessary to subdivide the chosen action into component parts and determine which organizations or units are to execute each part.

These outputs of the decision and direction function will provide inputs to the report generation function, resulting in the generation of directives and information that provide guidance for component actions.

b. Summary

Table C2-3 details the decision and direction function in terms of its inputs, processes, and resultant outputs.

TABLE C2-2. SITUATION ASSESSMENT.

Inputs	Processes	Outputs
1. Own force mission and task statement	1. Analyze mission and task to determine overall & physical objectives: Review mission statement, determine rationale for mission objectives Determine criticality of own tasks to success of overall mission Determine how own tasks support or are supported by other forces	1. Immediate & long-range effects 2. Statement of overall & physical objectives 3. Significant elements of problem 4. Contributions to superior's objectives
2. Own force characteristics: Platforms available Disposition Platform characteristics Weapons Sensors EW/Communications Readiness state Reconnaissance/surveillance capabilities Performance capabilities	2. Determine success criteria related to own mission	2. Criteria for selecting among tentative COAs: Suitability Feasibility Acceptability
3. C^3 structure; own & enemy Chain of command Forces Communications (interconnectivities) Major responsibilities/capabilities	3. Review forces assigned to assist in accomplishing own tasks: Determine general composition of forces involved in mission Determine operational readiness/ requirements of own forces	3. Deficiencies in force composition and operational status

TABLE C2-2. (Continued)

Inputs	Processes	Outputs
4. Characteristics of operating area: General political, economic, sociopsych factors Environmental Geographic Climate/weather Times of transportation/supply Fixed facilities Time/distance factors	4. Survey location & composition of enemy forces to identify potential threats to missions: Identify force concentrations/buildups Identify fixed and mobile sources of support	4. Disposition of potentially threatening forces
5. Enemy force characteristics: Platforms available Dispositions Platform characteristics Weapons Sensors EW Communications Performance capabilities	5. Derive relevant enemy capabilities by considering enemy force disposition, composition, and characteristics together with environmental aspects of the operating area	5. Overall capabilities and weaknesses of enemy forces
6. Intelligence Estimate: Enemy objectives/intentions Current state of readiness & combat efficiency Reconnaissance/surveillance capabilities Fixed/mobile support	6. Generate alternative enemy COAs to achieve objectives: Formulate means of implementing enemy capabilities Assess the ability of supporting forces to sustain specific COAs	6. Plausible spectrum of enemy COAs

TABLE C2-2. (Continued)

Inputs	Processes	Outputs
7. Intelligence estimate: Enemy force disposition, mobile and fixed Enemy sensor & weapon capability Enemy orders of battle Environmental data	7. Screen enemy COAs against indicated intentions to identify those most probable; estimate reaction to own mission	7. Probable enemy COAs
8. Mission data: OPORDERS/directive Mission objectives Rules of engagement Own force characteristics Force composition/disposition Sensor/weapon capabilities Readiness Characteristics of operating area Geopolitical aspects Environmental data Emcon considerations Time/distance factors	8. Identify capabilities and limitations affecting own possible COAs. Assess the effects due to nature of the physical objectives, forces available, readiness and disposition, environmental conditions, and rules of engagement *Determine the implications of specific* COAs adopted by the enemy	8. Own strength and weakness factors
9. Own & enemy force characteristics: Compositions/dispositions Readiness Sensor/weapon capabilities Characteristics of operating area Environmental aspects Enemy site locations Emcon policy Intelligence estimate Predicted/expected tactics	9. Compare own & enemy strengths and weaknesses: Formulate assumptions & conclusions concern-relative strengths and weaknesses of opposing forces Identify enemy strengths to be avoided, weaknesses to be exploited	9. Preliminary assessment of own force's ability to accomplish assigned mission

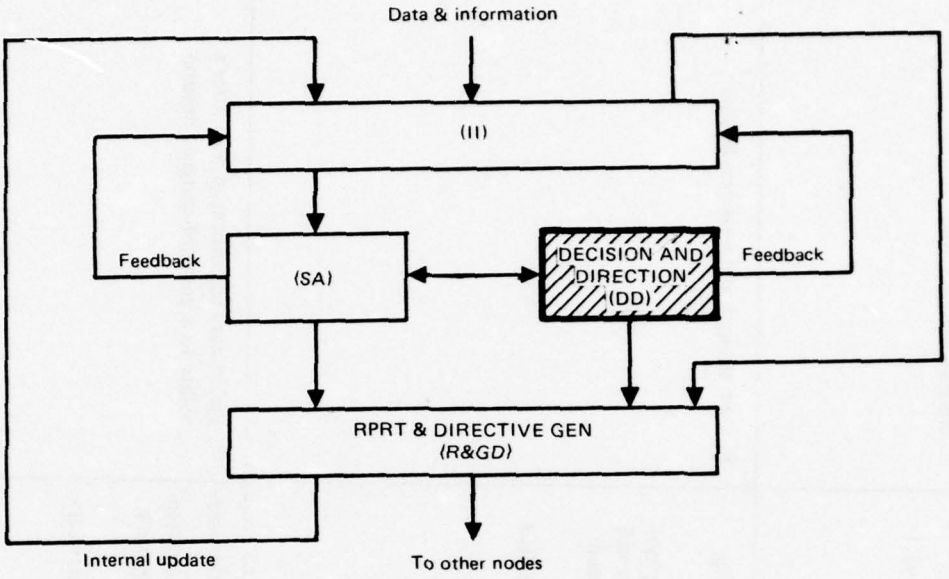


Figure C2-9. Decision and direction flow diagram.

C2.6.4 REPORT AND DIRECTIVE GENERATION FUNCTION

Having selected a particular course of action and derived the associated tasking and force assignments, there remains to communicate the specifics concerning its execution as indicated by figure C2-10. Such resource allocation information and supporting guidance, together with situation updates, alerts, and query responses from ongoing operations, constitute the principal elements to be transferred to external commands.

a. Scope

All processes involved in acquiring, structuring, formatting, and distributing information from internal sources destined for both internal and external addressees fall within the province of report generation. The function is initialized by establishing reporting responsibility thresholds – cues to create specific reports to cognizant superiors, coordinates, and subcoordinates at predetermined times or upon the occurrence of predetermined events.

When such a threshold is reached, the process of information acquisition and preparation for distribution begins. Information is extracted from the data base, structured and formatted in accordance with addressee requirements, and prepared for transmission. Assuming that transmission media are available, final disposition is made externally and/or internally as a data base update. The receipt of a proper acknowledgment terminates the function.

While the foregoing qualitative description is appropriate in a generic sense, the actual implementation of the report generation function is highly situation-dependent. The time required to produce and distribute a given report necessarily varies with its content, format complexity, and length, but in all instances the overriding consideration is that delivery be effected promptly, allowing the recipient time to take appropriate action. This

TABLE C2-3. DECISION AND DIRECTION.

Inputs	Processes	Outputs
1. Concepts & conclusions derived from previous summations and generated in situation assessment. 2. Previously employed own & enemy strategies/ tactics 3. Current ROE, directives	1. Formulate tentative own COAs including force compositions to meet mission objectives: Conceptualize appropriate actions, within the capabilities of own forces, which would achieve mission objectives Define, as a tentative COA, each distinct sequence of actions	1. Tentative COAs
1. Tentative COAs 2. Current disposition of own & enemy combat and supporting forces 3. Statement of anticipated intelligence requirements	2. Determine preliminary requirements of each COA: Examine problems involved in positioning, sustaining, defending own forces pursuing a particular COA Determine how the actions will be carried out, when, size & composition of forces involved, amount of time employed Define requirements for offensive/defense actions, positioning & movement, lotistics, intelligence	1. Preliminary offensive/defensive action, coordination reqt, logistic, intelligence, force composition reqt
1. Tentative COAs, intelligence & logistics requirements 2. Own force characteristics, support capabilities 3. Preliminary retained COAs	3. Evaluate each tentative COA for: (a) Suitability Determine whether each COA can produce the minimal effects necessary to meet success criteria. List associated contingencies & assumptions (b) Feasibility Determine whether each COA can be accomplished with available forces, supporting elements (c) Acceptability Determine probable outcomes of opposing COAs	1. Retained suitable COAs (or) 2. Direction to an entry point elsewhere within node 3. Retained suitable and feasible COAs (or) 4. Direction to an entry point elsewhere in node 5. Estimated losses for suitable and feasible COAs 6. Preliminary acceptability assessment

TABLE C2-3. (Continued)

Inputs	Processes	Outputs
4. Enemy COAs	Game each tentative COA vs each enemy COA to determine probable interactions & outcome Estimate own losses for each combination of opposing actions Assess whether the probable results justify the estimated costs	7. Retained COAs (or) 8. Directing to an entry point elsewhere within the node
1. All retained COAs 2. Acceptability assessments	4. Review conclusions established in previous analyses; determine & consider advantages/disadvantages of each retained COA	1. Advantages/disadvantages of each retained COA
1. Evaluated & retained COAs	5. Select the COA which meets established success criteria with minimal expenditure of resources	1. Selected COA
1. Selected COA 2. Doctrine, Experience, ROE	6. Determine individual operations needed to accomplish chosen COA	1. Tabulation of critical events, timing considerations, requirements for coordination 2. Individual operations associated with chosen COA
1. Characteristics of the OP area 2. Tabulation of critical events, timing, considerations, requirements for coordination 3. Individual operations associated with chosen COA	7. Determine how the characteristics of the physical objectives and environmental factors will influence the execution of the chosen COA	1. Implication of OP area characteristics on chosen COA
1. Disposition & characteristics of enemy forces in OP area 2. Estimate of enemy capabilities 3. Implications of OP area characteristics 4. Historical file of enemy strategies	8. Enemy Strategic & Tactics: (a) Determine the composition of enemy forces most likely to oppose own forces in each individual operation (b) Review enemy reactions and strategies pursued in previous operations of a similar nature	1. Probable enemy tactics

TABLE C2-3. (Continued)

Inputs	Processes	Outputs
1. Probable enemy tactics 2. Individual operations 3. OP area characteristics	9. Considering safety of own force & minimal resource expenditure, develop counteractions to most probable enemy actions, compatible with own component operations	1. Counteractions to be employed vs probable enemy COAs
1. Disposition, composition, status/readiness of own forces 2. Individual operations, actions, & counteractions	10. Determine resource allocations for each individual operation	1. Tasking for own combat & supporting forces

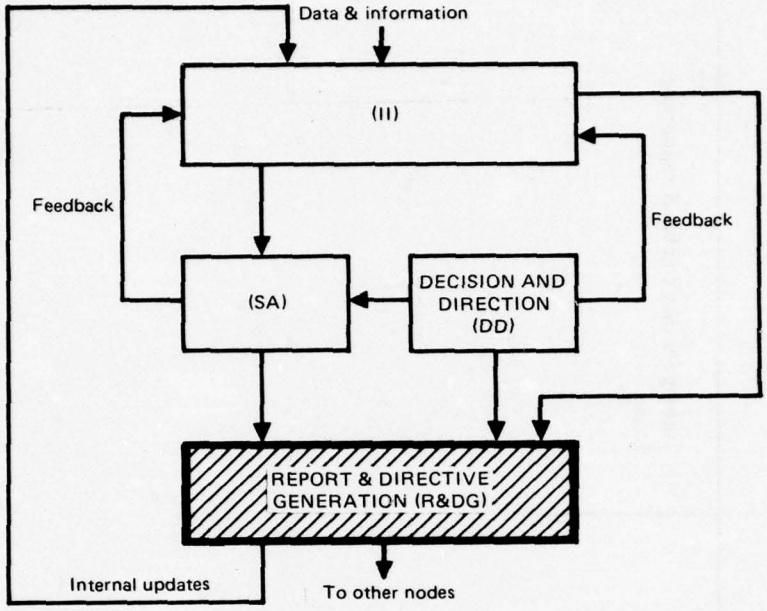


Figure C2-10. Report and directive generation function flow diagram.

consideration may in fact be of such importance as to effectively cause the bypass of those generic subfunctions judged as not making a sufficient contribution to warrant their performance.

Accordingly, the sequence of inputs, processes, and outputs associated with report generation, like the sequence of the previous functions, must be viewed as appropriate only under the assumption that sufficient time can indeed be allocated to their performance.

b. Summary

Table C2-4 details the report generation function in terms of its inputs, processes, and resultant outputs.

TABLE C2-4. REPORT AND DIRECTIVE GENERATION.

Input	Process	Output
1. Own force tasking, chain of command, organization 2. Reporting requirements to superiors, coordinates, and subordinates. 3. Physical & environmental factors 4. Probable enemy COAs	1. Establish report content reqts and reporting thresholds (implicitly contains primary addressees) for both external reporting and internal data base update	1. Reporting/updating thresholds and requirements
1. Reporting/updating thresholds and reqt 2. Operational reports, internal & external 3. Internally generated planning phase products 4. Ad hoc queries	2. Determine when thresholds have been exceeded	1. Cue for specific report creation
1. Report generation/data base update cue; content requirements 2. Operational situation data, alerts 3. Resource allocation information and supporting guidance 4. Data base query responses	3. Determine final internal and external distribution and routing	1. Specification of distribution and routing, report structure, and formatting
1. Specification of distribution and routing, report structure, and format table	4. Determine transmission media, retrieve information, and format appropriately	1. Formatted report
1. Formatting report	5. Transmit report to internal and external addressees	1. Transmitted report
1. Transmitted report	6. Generate internal/external acknowledgment	1. Report receipt acknowledgment
1. Report receipt acknowledgment	7. File receipt acknowledgment	1. Record of receipt

C2.7 MAN-MACHINE COUPLING

Man-machine coupling functions have been classified simply as those corresponding to the human senses as shown in figure C2-11. Another category of man-machine interface is the developing area of coupling by means other than the human senses such as direct measurement of electrical output of the brain or electrical stimulation of the brain nerve system.

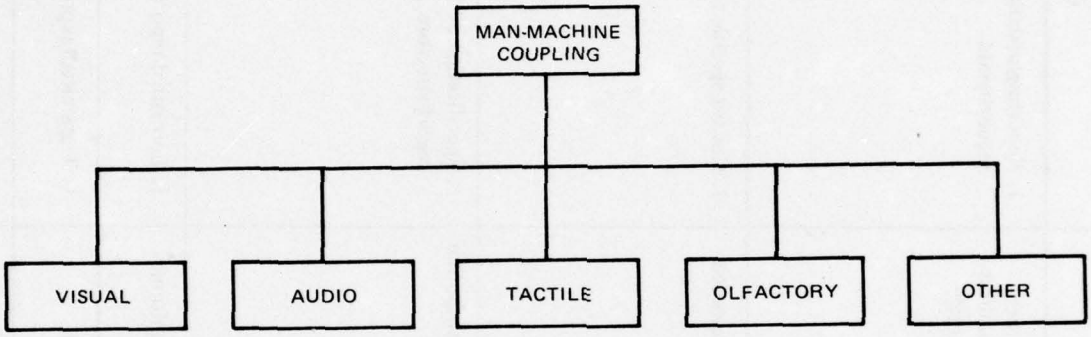


Figure C2-11. Functional breakdown of man-machine coupling.

C3.0 SUBSYSTEM STRUCTURE OF A C^3 NODE

Section 2 described the functional structure of a node; that is, presented a structured checklist of what is done in a node.

This section describes the subsystem structure of the generic C^3 node; ie, presents a structured checklist of what the designer must construct or create to perform the functions. Figure C3-1 shows the five major subsystems of a node and the elements of these major subsystems.

The reader will note that this is a significant departure from the traditional way in the Navy of classifying systems by names such as radar, sonar, and ESM. The structure given here is fully consistent with classical system engineering, however, and most important, is configured (as the traditional structure is not) so that, in a hierarchical distributed network, the system designer can rigorously define the boundaries and input, output conditions of every subsystem. Rigor in defining boundaries and input/output is essential, and the system structuring concept has been driven by this requirement.

The major subsystems are described in terms of their elements in the following sections.

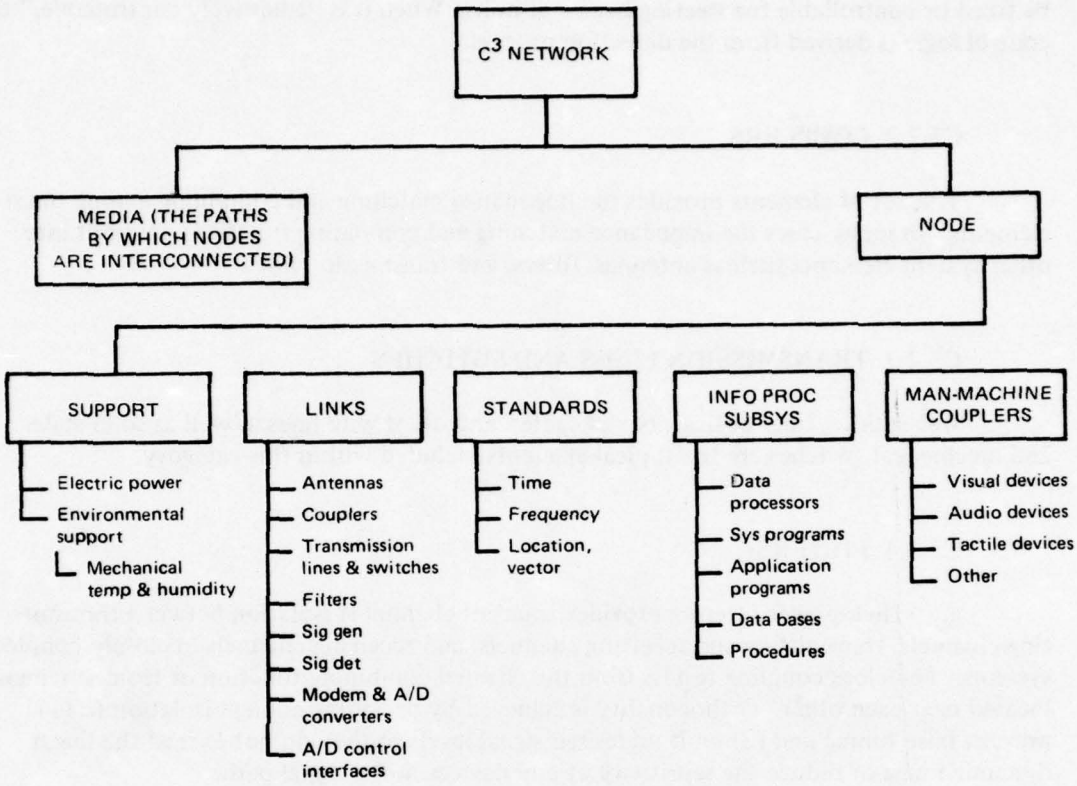


Figure C3-1. C^3 system/subsystem breakdown.

C3.1 SUPPORT SUBSYSTEM

The electronic system support element includes the electric power equipment, the temperature and humidity control equipment, and the structural supports and foundations.

C3.2 LINKS

The transmission/reception link subsystem is the analog signal processing element of the node. Figure C3-1 shows a breakdown of the transmission/reception subsystem (LINKS). Different subsets of these elements are selected as appropriate for different purposes, platforms, frequencies, and electromagnetic environments. A brief description of each element follows.

C3.2.1 ANTENNA

This element provides radiation or capture of electromagnetic energy into or from space and includes the radiation element(s), the tuning or matching element which provides the impedance matching between antenna and transmission line, and the radiation element combiner which provides combining of antenna elements into arrays. The combining may

be fixed or controllable for steering beams or nulls. When it is "adaptively controllable," the control logic is derived from the detection process.

C3.2.2 COUPLERS

This set of elements provides the impedance matching and combining among the rf elements. In many cases the impedance matching and combining functions are built into other system elements such as antennas, filters, and transmission lines.

C3.2.3 TRANSMISSION LINES AND SWITCHES

Waveguides, fiber optics, coaxial cables, and other wire lines as well as solid-state and mechanical switches are the typical elements included within this category.

C3.2.4 FILTERS

a. High-power filtering provides adjacent-channel rf isolation between transmitting channels, transmitting and receiving channels, and receiving channels in closely coupled systems. The close coupling results from the channel combining function or from antennas located near each other. Orthogonality is achieved by providing enough isolation to (1) prevent false tuning and (2) limit undesired signal levels so they do not exceed the linear dynamic range or reduce the sensitivity of any devices in the signal path.

b. Low-power filtering is used for situations in which close coupling creates a requirement to provide additional isolation of sensitive detection devices from (1) adjacent-channel transmitter power and (2) out-of-band noise generated in rf synthesis and generation elements.

C3.2.5 SIGNAL GENERATORS, DETECTORS, CONVERTERS

a. Signal synthesis is used (1) to generate reference signals for detection processes and (2) to generate information carrying rf signals for radiation into space either with or without additional power amplification. The information impressed on the rf signal is derived from the modem.

b. Power amplification allows various levels of rf power generation capability.

c. Detectors and converters amplify a weak received signal and convert it in frequency (either to an intermediate frequency or to baseband). If the conversion is to baseband, this may include intermediate-frequency (if) and/or baseband signal amplifiers, filters, etc.

C3.2.6 MODEM AND A/D CONVERTERS

This element modulates and/or demodulates signals and provides the interface between rf generation and detection devices and the baseband signal processing devices. It

may include baseband and if amplifiers and signal conditioners (eg, a typical interface to rf detection and generation devices is via an intermediate 70-MHz information carrying signal). The postdetection combining element is typically used on fading signals or bursty channels. It combines two or more channels, carrying the same information, in such a way that the information content is more accurately reconstructed. The logic may be based on both "hard" and "soft" decisions and requires a degree of decorrelation of the degrading conditions affecting the different channels to be combined. Keystream interfaces for use of crypto sequences in AJ, AI modes are also included.

C3.2.7 A/D CONTROL INTERFACES

These devices provide the interface between devices which are typically analog (electronic or mechanical – eg, tuning shaft positioning) in the transmission subsystem and the typically digital control and monitoring signals from or to the data processors.

C3.3 STANDARDS

This subsystem provides the reference signals for own platform time, frequency, and platform location and motion vectors. Devices such as cesium, rubidium, and quartz clocks as well as compasses and gyros are included in this category. These devices are connected via the information processing and link subsystems to other nooks so that variations of individual standards at individual nodes can be adjusted on the basis of information from other nodes.

C3.4 SUBSYSTEM COMPONENTS

The information processing subsystem is composed of:

- Data processors
- System programs
- Data base
- Application programs
- Manual procedures

C3.4.1 DATA PROCESSORS

The data processors consist of hardware such as mini and micro processors and data storage devices.

C3.4.2 SYSTEM PROGRAMS

The system programs provide the logical basis for the operation of the data processors. These programs include:

- Multiprocessing, multimode time-sharing programs for transition-oriented, packet-switched processing
- Scheduling and networking including access, error control and queue management
- System monitor, trend analysis, system control, system initiation, and diagnostics
- File management
- Man-machine couplers support programs

C3.4.3 DATA BASE

The data base is the totality of information stored in the data processors to support both system and application programs. Major data base types include:

- System
- Tactical/strategic
- Logistic
- Administrative

A sample description of a data base is given in appendix F.

C3.4.4 APPLICATION PROGRAMS

These are the user-oriented programs which allow the system subscribers (eg, Tactical Action Officer) to use the system in ways tailored to their operational needs. Examples of such programs are:

- Action recall and replay
- Navigation
- Threat assessment
- Data base access
- Gaming
- Multisource
- Information correlation
- Event-driven warnings at set thresholds
- Information display, input

C3.4.5 MANUAL PROCEDURES

Procedures are the instructions for operation of the C^3 system by people. This includes procedures for both efficient operation of the electronic system and efficient, effective interaction (man-machine and man-man) for performing C^3.

Procedures can be classified into areas such as:

- System operating instructions
- System maintenance instructions
- Organization and recognition of tactical information
- Quantification of tactical values such as outcome preferences, risk attitudes, probability, and judgment
- Conceptualizing tentative courses of action (COAs)
- Establishing suitability and acceptability criteria and making evaluations of COAs
- Methods of war gaming
- Generating orders and reports

C3.5 MAN-MACHINE COUPLERS

This category of subsystems includes

(a) Visual devices such as individual and group view displays, hard-copy readout, and other visual indicators such as light and flags

(b) Audio driven or generating devices such as earphones, speakers, microphones, and buzzers

(c) Tactile devices such as keyboards, buttons, and footpedals

(d) Other devices which electrically stimulate the nervous system or brain directly or which directly derive information from measurement of brain electrical activity.

C4.0 ARCHITECTURAL CONSIDERATIONS

In this section some considerations are discussed which lead to the selection of a distributed microprocessor/microcomputer architecture for the future NC^3N.

C4.1 SPECIFIC CONSIDERATIONS

C4.1.1. ECONOMY

Cost reduction or life-cycle cost minimization, while not a functional requirement per se, is a major consideration in the post-1985 era. All aspects must be considered in order to achieve the most cost-effective design. Modularity, automation/manpower

reduction, and standardization are some design criteria to be considered. In addition, the use of commercially available equipment is important to this objective. Recently, manufacturers have started to produce commercial equipment that satisfies military specifications; this trend should be firmly established and encouraged by DoD.

C4.1.2 SURVIVABILITY

Survival of nuclear and radiation effects is a requirement of all NC^3N equipment that must be available throughout a nuclear conflict. This includes the equipment in airborne platforms, surface platforms, and other force-control or autonomous systems.

Two types of nuclear environments are involved: the target and nontarget environments. In the former, the equipment would probably be destroyed. In a nontarget environment, the threat would be to operational connectivity and to the information; collateral effects of nuclear explosions, while not damaging the equipment, may cause temporary malfunctions that may destroy information.

In directly targeted equipment, the primary requirement is for correct operation as long as the platform is not destroyed. For equipment in the nontarget environment, special devices or procedures must be implemented to protect the system from collateral nuclear effects.

C4.1.3 FLEXIBILITY/ADAPTABILITY

Any system as widely deployed as the NC^3N must be capable of providing a long, useful service life if its total cost is to be realistic. This will be the case only if the system, once installed, can evolve gracefully to keep pace with newly emerging technologies and techniques. During the service life of a given platform, new systems will periodically be installed, and its operational assignments modified. With the passage of time, the configuration aboard an existing platform will need to expand in capability as new technology is introduced. Trends toward multifunction data links will greatly modify the architecture of existing systems such as NTDS, and the NC^3N must continue to provide its support functions in the interim time frame. Introduction of new weapons may greatly increase the required volume and nature of the information that the platform must exchange with other nodes.

None of these goals may be achieved cost-effectively if wholesale disruption of the existing NC^3N is required to achieve incremented expansion, or if such change requires addition of excess, unusable capability. On the other hand, a flexible, modular system architecture will allow the system to remain viable and up-to-date even in the face of major, unforeseen technological developments. The level of hardware and software modularization proposed, and its close correspondence with identifiable NC^3N functions, must ensure that technological adaptability will be an inherent system feature.

Nodes must be able to coordinate their actions in operational time. The dedicated channelization which presently hampers EW and communications is a result, in part, of the present lack of internodal coordination capability. As the NC^3N takes on the characteristics of a highly survivable switched system, disciplines and techniques for adaptive management are required. Efficient procedures for such functions as transfer of net control in the event of node failure, interchange of supervisory information on established links, and emergency reconfiguration of networks and sensors will be beyond human capacity and must be automated.

The post-1985 NC^3N must reduce node manning requirements and provide means to perform a large volume of routine and highly complex tasks. To achieve these objectives, present operator tasks must be automated.

C4.1.4 VULNERABILITY

Naval platforms operate in hostile electromagnetic environments, and are subject to jamming, intercept, adverse weather conditions, man-made interference, and equipment problems related to the harsh shipboard environment. As a result, circuits are highly vulnerable to failure, often when most needed. The C^3 system design must alleviate this problem. The use of powerful error control coding modes, data interleaving techniques, and automatic rechanneling must make it possible to initiate action before traffic is interrupted. New equipment and frequency selection must be made automatically and coordinated among the nodes. Each must rapidly reconfigure to the agreed-upon channel and mode to allow continuous information flow.

Modularity and standardization over the entire population of naval platforms are required to reduce the volume of information given up about platform identities and tactical arrangements by unique radiations for communications, target detection, tracking, and ECM.

C4.1.5 INTEROPERABILITY

As an integral part of the total C^3 system, the node must be compatible with other elements and equipments on present Navy platforms and post-1985 Navy platforms. This requirement applies to current, interim, and post-1985 equipment and C^3 electronic elements both internodal and intranodal. The NC^3N node must also be interoperable with other services.

Standardization is extremely important in the nodal design in order to meet the requirements of compatibility and interoperability. Standardization is also needed in order to provide an orderly, cost-effective transition and to meet the needs of expandability and contractability for different platform types.

Standardization is particularly important in the area of hardware-software interfaces. A variety of computer architectures is available for meeting the various computing-speed and memory-hierarchy requirements. However, it is necessary to apply some architectural controls in order to ensure total utility, flexibility, interoperability, and growth of the system. Among the design principles that can be applied to achieve these elements are (a) standardization on the microinstruction level, symbolic machine instruction level, or higher-order language (HOL) level, thereby ensuring that the system hardware complexities remain transparent to the software packages and that various architectures can be substituted with minimal impact on existing software; (b) either one-for-one substitution of computer components or the addition of plug-in types of modules to provide flexible throughput growth; and (c) expandability of standardized data storage facilities.

C4.1.6 AVAILABILITY/RELIABILITY/MAINTAINABILITY

In many operational situations, unavailability of a functional capability is intolerable, even for short periods. Loss of data due to intermittent failures may also severely degrade system effectiveness. High availability implies the capability of reconfiguring systems. In general, this also includes different, perhaps manual, backup procedures.

Availability, reliability, and maintainability are directly affected by component and packaging technologies, circuit and subsystem design philosophy, and system architecture. For example, interconnections on circuit boards or on integrated circuit chips have been a major source of failures; with LSI techniques, however, the number of interconnections is drastically reduced and reliability is improved. System architectures incorporating redundancy can improve the effective reliability by masking failures.

Lower costs of LSI hardware permit increased use of redundancy and self-testing, and limited amounts of self-repair in computer systems at the present time. The trend will be stronger in the 1980s.

C4.2 BASIC DESIGN CONCEPTS

The basic design philosophies available for the NC^3N node fall into two categories – centralized and distributed processing. Under these two basic design concepts, several options are available. The options include the following elements and combinations of them:

Large-scale computers

Medium-scale computers

Minicomputers

Microcomputers/microprocessors

Hardware

Hardwire

Firmware

Software

C4.2.1 CENTRALIZED VS DISTRIBUTED PROCESSING

The most elaborate computer systems in use today are the large real-time systems which have been built up at great expense by the government and major industrial corporations. They cover applications such as military command control, airline reservations, time-sharing, on-line banking, and stock brokerage. Despite the fact that many of these systems have taken thousands of man-years of effort to bring to completion, the results mainly have not equaled expectations. To achieve the desired throughput, users have frequently had to substitute larger and larger computers as the complexity of software pushed up the overhead. In addition, successful systems have found that the increasing demand for service has outstripped the rate at which hardware and software can be expanded.

This type of growth cannot go on indefinitely, and it is now apparent that the only practical solution lies in segmenting the problem through networks of smaller computers to handle a distributed processing load. An interconnected set of small computers can be organized to provide more processing power than could possibly be built into one central supercomputer.

The distributed computing network brings many advantages which make it attractive once the basic operating principles have been established. The NC^3N can be made tolerant to failures, since the failure of one part of a distributed network of processors or processing elements can be made to have limited, or even zero, effect on total NC^3N operation. It is

fundamentally simpler to develop such a system since it can be built, tested, and put into service one leg at a time. It is not necessary to have a huge software complex running correctly before any service can be provided.

An interconnected computing system, therefore, has the ability to grow fairly easily as additional demands are placed on it. It can accept local modifications and improvements as special needs become apparent without impacting other parts of the node or network, provided the fundamental intercommunication rules are obeyed.

When further compared against the design constraints identified in section C4.1, the choice of a distributed processing architecture is obvious. Aside from the functional design drivers, the choice of a centralized processing architecture would force a needlessly high design and life-cycle cost; ie, separate hardware and software designs for various platform types and high redesign costs for design changes due to dynamic growth.

C4.2.2 CPU TRADEOFF

The central processing unit (CPU) options available for the node are identified in section C4.2.1. This section evaluates these options. It is needless to develop a tradeoff analysis relative to large-scale or medium-scale computers; the choice of a distributed processing architecture effectively eliminates either as an option. This CPU tradeoff, therefore, will be between a minicomputer-based and a microcomputer/microprocessor-based processing architecture.

A system of minicomputers has the ability to easily grow or contract with varying platform or site requirements. The same may be said for the microcomputer/microprocessor-based system. The advantage here lies with the latter in that sizing is dynamic. A microcomputer/microprocessor-based system has the capability of being completely modular, even to the extent of having microcomputer interfaces to users, devices, and links. The minicomputer-based system is limited to a step function and to hardware, hardwire, or software intervaces except for those that are very complex. Microcomputer/microprocessors may also use hardware, hardwire, or software interfaces; therefore, this is not a tradeoff factor.

A system of minicomputers can be made tolerant to failures, since the malfunctions of part of the system can be made to have limited, or even zero, effect on system operation. Again, the same may be said for microcomputer/microprocessor-based systems, and again the latter has the advantage. The extremely low cost of microcomputer/microprocessor elements allows greater redundancy and therefore a better "self healing" design factor. This same factor, therefore, provides better reliability, availability, and maintainability.

It is easy to phase a minicomputer system into current communications systems, since implementation can proceed one portion at a time. It is even easier to phase a microcomputer/microprocessor-based system into current communications systems, since implementation can proceed one module at a time.

Minicomputer hardware costs are going down; microcomputer/microprocessor hardware costs are also going down. Current hardware costs would probably be comparable. The rate of hardware cost decline is much greater for microcomputer/microprocessors; also, the completely modular aspect of a microcomputer/microprocessor-based design provides this system a much greater potential for cost-effective design and operation.

Initial minicomputer software costs differ only slightly from those of large machines, once compilers and operating systems have been developed. This cost is quite high, however, and a considerable amount of redesign is necessary to incorporate a system design change.

On the other hand, operating and executive software systems and support software are available for minicomputer-based systems. Software designs and support software are not generally available currently with microcomputer/microprocessors. This presents a small development risk for a microcomputer/microprocessor-based design. This risk is extremely small considering current trends and could be eliminated by applying near-term Navy R&D dollars in this area.

A key advantage of a microcomputer/microprocessor-based system is that after a standard set of software is available, a system software design can become a modular design function.

Minicomputers are becoming less expensive every year while their capabilities are rapidly approaching those of larger machines. This makes it economically feasible to automate even those sites/platforms not yet considered. Again, the same may be said for microcomputer/microprocessors.

As software technology is catching up in the minicomputer field, the trend is to more flexible software at a time when large computer operating systems become more complex and rigid. The advantage lies with the minicomputer-based systems as discussed above. The development risk is small with regard to microcomputer/microprocessor software. Higher-order languages are available for minicomputer-based system design. This is again a development risk for a microcomputer/microprocessor-based system.

Shipboard and airborne installations mandate at least a partially MIL SPEC'd computer; this is possible with a minicomputer-based system – and even practical with a microcomputer/microprocessor-based system.

Software for minicomputers can be partitioned in a manner such that the system can be expanded or contracted, building-block style, by adding/deleting computers. A microcomputer/microprocessor-based system has potentially much greater modularity. Standardized packages and support software must first be developed.

Minicomputers and microcomputer/microprocessors offer a powerful advantage to the programmer in microprogramming. Microprogramming enables tailoring of special software instructions not otherwise available. While a microcomputer/microprocessor-based system provides a microprogramming capability to a much greater degree of modularity, a minicomputer-based system has a much larger instruction repertoire.

While minicomputers surpass large-scale computers in maintainability and operability because of their inherent greater simplicity, microcomputer/microprocessors surpass minicomputers in these respects for the same reason. The low cost of minicomputer/microprocessor elements makes it practicable to replace a malfunctioning microcomputer/microprocessor and throw away the malfunctioning unit.

Although price comparisons are sometimes misleading, it is fair to say that an LSI microprocessor has a substantial cost advantage over a typical minicomputer CPU. For example, a complete LSI CPU may be purchased for as little as $300, compared to $1000 to$20 000 for a minicomputer CPU.

The CPU power consumption of an LSI microcomputer is 66–75% less than that of a comparable minicomputer. For a system containing but one CPU, the difference would not be significant, considering the overall system power requirements. However, in applications in which many CPUs are required, the power difference would be substantial.

An MOS/LSI microcomputer operates at 50–33% of the speed of commercially available minicomputers. Typical memory-to-memory add times for a moderately priced minicomputer are 5–20 microseconds compared to 15–60 microseconds for a microcomputer. The speed of a microcomputer is derived from the particular MOS process used in fabrication. As these processes improve, so will the speed.

With integrated circuits, system reliability is largely a function of the number of printed circuit board interconnections. Since each LSI package replaces from 50 to 100 TTL packages, the interconnections required by microcomputers are reduced and total system reliability is increased. The LSI microcomputer can be built into a light and compact configuration because of the higher number of gates per package module and the simplicity of interconnection.

Finally, the LSI microcomputer offers better price-performance, lower power consumption and heat dissipation, higher reliability, and smaller physical size than a minicomputer. The microcomputer further offers greater flexibility of microprogramming, which, in a given application, has many advantages. Although execution speeds comparable to those of today's minicomputer have not yet been achieved, several architectural techniques have emerged which substantially increase LSI microcomputer capacity and speed. In 1985 such speeds and capacities are expected to be equivalent to those of current host computers.

Considering the above tradeoff parameters in terms of the design drivers (and cost-effectiveness criteria), a distributed processing architecture based on microprocessor/minicomputer building-block elements is recommended.

C4.2.3 SYSTEM ARCHITECTURE

Based on the discussions in the previous sections, the logical system architecture for the NC^3N has been determined to be a distributed processing architecture using microprocessors/minicomputers as basic modules. The following sections discuss: (a) basic considerations; (b) application in a distributed processing concept with respect to the design drivers discussed in section 4.2.1; (c) some problem areas related to the use of microprocessors and microcomputers as basic modules for the NC^3N node; (d) and a baseline architectural design concept using microprocessors and microcomputers as the basic modules in a distributed processing framework.

C4.2.3.1 MICROPROCESSOR/MICROCOMPUTER BASICS (AS OF JUNE 1975).

The distinguishing characteristic or component of a microcomputer is the microprocessor, one or more large-scale-integration (LSI) chips that perform the basic functions of a processing unit. Contained within a typical 0.16-inch-square package (the basis for the "micro" designation) are the usual elements of any processor – the arithmetic logic unit, I/O control logic, and general-purpose registers. When memory and a complement of I/O devices accompany or work jointly with a microprocessor, a microcomputer is formed.

Present microcomputers incorporate devices fabricated by metal oxide semiconductor (MOS) techniques. MOS devices offer extremely high densities of transistors per unit area, but are inherently slower than bipolar devices. Current MOS speeds for a logic element or chip range from 40 ns for fast, n-channel silicon gate devices, to 200 ns for p-channel metal units. Architectural attributes which exploit MOS technology have been added to increase the speed of microcomputers vis-a-vis bipolar units. They consist of hardware index registers, parallel bus structures, register stacks with programmable stack pointers, and decimal arithmetic.

The microcomputer is the apparent successor to the minicomputer as the latest and most advanced evolutionary step in EDP. At commercial introduction in 1965, minicomputers constituted a revolution in data processing. Their compact size and low cost

permitted the development of dedicated systems to meet specialized needs in communications, control, data acquisition, and small business data processing.

The potentials of minicomputers were at first not appreciated by system designers, who viewed minis unfavorably in terms of the features and programming languages offered by the larger machines. Program loading was awkward and time consuming, and the shorter word lengths and limited instruction sets made minicomputer programming tedious. Today, systems designers are more familiar with the vagaries and capabilities of minicomputers, and are implementing minis in a myriad of applications.

Microcomputers and microprocessors are following a similar course. In many existing minicomputer applications they offer improved price-performance, compactness, and reliability. Moreover, the characteristics of the LSI microprocessor lend themselves to new applications and system concepts that are impractical with minicomputers.

As the minicomputer evolved upwards into the high end of small-scale systems, electronic technology was advancing in circuit miniaturization and the use of MOS as a low-cost alternative to bipolar logic. This steady advance in MOS technology has increased the large-scale integration of digital circuits from 100 MOS transistors per chip to over 14 000 per chip during the last 5 years. This increase in chip density has caused a revolution in digital hardware applications. Among the more publicized are the pocket calculator and the digital watch.

A microcomputer uses no more than 10 MOS/LSI packages, each holding more than 500 transistor circuits. A minicomputer would typically require about 100 TTL packages. This simple comparison reflects the prime differences between a minicomputer and a microcomputer – physical size and complexity of components.

A concurrent development which has contributed to the evolution of microprocessors, and thus microcomputers, is microprogramming, in which each machine instruction initiates a sequence of more elementary instructions (microinstructions). A microprogramming approach allows replacing fixed, conventional CPU control logic with a control memory. Addresses in control memory represent unique states in conventional control logic, and each memory output represents control lines from conventional logic. Stored in this memory are basic microinstructions, including the fundamental control, testing, branching, and moving operations.

For an LSI machine to perform higher-level operations with ease, microinstruction sequences corresponding to common higher-level functions are stored in a separate read-only memory (ROM) to be accessed, decoded, and executed on command. These high-level sequences are called macroinstructions, the medium in which system programmers usually code. Macroinstructions in a microcomputer correspond to the basic instructions of a minicomputer. Microprogramming enables a systems designer to adapt standard hardware to specific applications – perhaps the most useful characteristic of a minicomputer. The designer can construct macroinstructions that are best suited for the particular functions to be performed, and incorporate them into the microprocessor. For example, the instruction set of an existing minicomputer can be completely or partially emulated to minimize software development. Alternatively, a machine can be built to perform functions peculiar to an application such as word processing or data acquisition. This capability to adapt a standard set of hardware modules to a variety of problems combines the cost advantages of high-volume chip production with the computing efficiency of tailored instruction sets.

More than 20 different microprocessors have appeared within the 4 years since the introduction of the Intel 4004. Most of the machines still use the PMOS technology, which has been developed to such a level that one 12-bit microprocessor has 11 000 transistors on a single 0.22 by 0.24-inch chip. Most of the newer microprocessors, however, use either the

NMOS technology, in which the transistors operate by means of negative current carriers (electrons), or the complementary MOS (CMOS) technology. CMOS combines the PMOS and NMOS technologies to achieve a reduction in power requirements and to improve resistance to extraneous noise. The first bipolar microprocessors made their appeearance in late 1975.

Coming 3 decades after the first electronic computers,, microprocessors benefited from the experience accumulated in system organization and computer architecture. Many advanced concepts and features, frequently unavailable in machines several orders of magnitude larger and more expensive, are standard in almost every microprocessor. One such feature, known as a stack, is a set of electronic registers organized so that the subroutines called up by a program are handled on a last-in, first-out basis: after executing one subroutine or more, the microprocessor can return quickly to the main program sequence.

The parallel development of many microcomputer systems has given rise to a number of original designs and architectures. "Architecture" refers to the organization of a computer. The everyday meaning, referring to both a style of construction and a particular way of assembling structural materials to achieve a functional goal, also carries over into the field of computer design. Computer architecture describes the arrangement of the CPU, the memory elements for the storage of programs and data, the input and output devices, and the master clock. Thus, one architecture may emphasize facility of arithmetic operations and another convenience of input and output operations. Whereas both have a CPU, a memory, and input and output ports, the first is more suited for lengthy numerical analysis and the second for control applications and the monitoring of external equipment.

The CPU, or microprocessor, is the most expensive component (or group of components) of a microcomputer. It fetches the control instructions stored in the memory and then decodes, interprets, and implements them. The CPU manages the temporary storage and retrieval of data and regulates the exchange of information with the outside world through the microcomputer's input and output ports. It incorporates the arithmetic and logic unit (ALU), in which all operations are performed, and a certain number of registers. Finally, it synchronizes the operation of the various components.

A microcomputer is usually classified according to the number of bits that can be handled by its CPU. Its performance is judged by the richness of its instruction set, by the bit efficiency of its program (the number of bits that need to be stored in the program for the implementation of a given set of tasks), and by the speed with which it executes typical programs. Such distinctions have mainly to do with capacity for operating in real time. If speed of operation is not a consideration, almost any microcomputer can serve in a given application. Some machines, however, may be more economical than others for particular jobs.

Four-bit, single-chip microprocessors are particularly economical. They are well suited to systems designed for decimal arithmetic, or to systems that do not have to deal with "words" consisting of many bits.

Eight-bit microprocessors are the most popular at the present time. Their word length makes them a natural choice for all applications that involve communications equipment, which commonly works with eight-bit encoded characters. They have more complete instruction sets and more computing power than the four-bit units and have many of the features found in larger machines. Although many of them require more supporting devices, some of the most recent eight-bit CPUs can make a complete minimal microcomputer with as few as five devices. A few 12- and 16-bit machines have been introduced, mostly for process control and other complex tasks. Some of them are highly integrated versions of previously available minicomputers, for which they are an economical substitute when speed is not critical.

Although the microprocessor is the most complicated and expensive single function of a microcomputer system, it is in fact completely controlled by the memory that surrounds it. In the evaluation of a CPU attention must be given to the variety of modes made available for accessing instructions and manipulating data, since these operations significantly influence the ease of programming, the speed of execution, and eventually the size of the memory itself. In many cases the most costly part of a system's hardware is the memory subsystem. For many years the memory requirements of digital systems were met with ferrite cores – tiny ceramic rings densely strung on a mesh of fine wires. Core memories are still widely used in many large computers and minicomputers, but an increasing number of users are turning to semiconductor memories. Fabricated on silicon chips by the same technology used to make microprocessors, semiconductor memories require less power than core memories, are easier to use, take up less space, and have recently become less expensive. Since MOS memories are also far easier to integrate into an MOS microcomputer system, they are the overwhelming choice of microprocessor users.

Most semiconductor memories found in microcomputers are of the random-access type. Access to any memory location can be gained in a uniform amount of time. The two main types of random-access memory differ in the "volatility"; that is, in their ability to retain their contents under various operating conditions. The computer program and tables of fixed data are mainly stored in read-only memory (ROM) devices. ROMs are nonvolatile: their contents cannot be altered during the operation of the computer, and the retention of stored data does not depend on a supply of power. The contents of an ROM are simply binary patterns of 1's and 0's that are programmed in advance by the user. When large volumes of identical ROMs are required, the programming is most economically done by the semiconductor manufacturer's custom-making one of the masks employed during fabrication. Such an ROM is said to be "mask programmed." Mask-programmed ROMs can store up to 16 384 bits, and the cost can fall to less than 0.1 cent per bit.

When only a few ROMs are to be programmed, as in laboratory development work, one can use "field programmable" ROMs (PROMs). A popular type of PROM, introduced almost concurrently with the first microprocessors, can be programmed and then erased. It incorporates arrays of floating-gate avalanche-injection transistors, which are capable of trapping a charge when a pulse in excess of 40 volts is applied to them. (Microcomputers, depending on the technology used in the chip, operate at from 5 to 17 volts.) The charges can subsequently be removed by exposing the device to intense radiation (ultraviolet or X rays). The trapping of charge is what makes it possible to program the device; the removal of charge is equivalent to erasure. The cost per bit for PROMs is about 10 times more than for mask-programmed ROMs, and the capacity of the devices is at present limited to 8192 bits.

The temporary storage of data calls for memories that can be modified while the microcomputer is operating. This capability is found in the semiconductor "read/write" memories called RAMs. The province of read/write memories is currently one of the most competitive of the semiconductor industry. Typical devices can store as many as 4096 bits at costs that are dropping toward 0.1 cent per bit.

A typical microcomputer system incorporates both volatile and nonvolatile memories. Some applications, however, do not need temporary data storage beyond what is provided by the microprocessor's internal registers, so that the read/write memory can sometimes be omitted. In other applications permanent storage is omitted and read/write memories serve to store both the program and the data. When only read/write memories are employed, the program must be reloaded each time the power is shut off (either on purpose or accidentally) unless a battery backup has been included in the system to

maintain the contents of the memory. With some of the newer CMOS semiconductor memories such a backup system can operate for weeks on as few as three or four penlight batteries.

Currently under development are all-electronic bulk-storage systems that will use charge-coupled devices or magnetic-bubble memories. They should be cheaper, faster, more capacious, and more reliable than present systems.

Hardware support is available in the form of completely assembled subsystems. Fabricated as boards or modules, the subsystems can be directly incorporated into larger systems of many kinds. The availability of standardized but programmable modules is particularly useful for prototype development or for equipment manufactured in small series, which do not justify a custom-made microcomputer design.

It is well known that in any computer system the software is likely to account for the largest fraction of the development cost. This is no less true for microcomputers, and it is indicative of the importance of giving the user comprehensive support. Although few microcomputer software packages compare favorably with those available for larger and more expensive machines, they considerably simplify the task of putting together a microcomputer system.

A microcomputer program consists of a sequence of binary words stored in a control memory. The instructions thus defined are said to be written in machine language. Although a programmer can elect to write his program directly in this form, the process is time-consuming and prone to error. Programming is made considerably simpler by assembly languages, which are available for all microcomputers. These languages allow the substitution of mnemonic words such as ADD, SUB, and JUMP for the binary words of the machine language; they also simplify the task of putting program data into a memory by giving the memory "addresses" arbitrary labels instead of absolute locations. An assembly language program must be translated into machine language before it is committed to a memory; this conversion is accomplished by an assembler, which checks the assembly language program for certain types of errors and, if none are found, produces the desired machine language code. The assembler is a program that sometimes can be executed by the microcomputer itself.

A higher level in the hierarchy of programming languages is represented by procedure-oriented languages, such as Fortran of PL/M. To translate statements written in these languages into machine language, one uses a compiler. When a compiler is available, it speeds up programming, and the resulting easier-to-understand programs simplify the problems of documentation and maintenance. Unfortunately, compilers, because of their general nature, tend to generate machine language programs that are not highly efficient in speed of execution or in number of instructions. Typically they require from 10% to 100% more control memory than would have been needed if the programmer had worked at the assembly language level. The decision whether or not to use a compiler is based on tradeoffs involving the experience of the programmer, the time available for software development, and, probably most important, the expected production volume of the system. Economies in memory achieved by more efficient programs can obviously justify greater software development costs.

The demand for microcomputers and the continuing evolution of solid-state technology will reduce the cost of microprocessors and memories, will improve production yields, and will lead to higher levels of integration. Particularly promising at the present time is one of the newest bipolar technologies – integrated injection logic (IIL). This technology is characterized by a greater density of components on the substrate, higher speed of operation, and lower power requirements. Microprocessors will be among the first devices to exploit these advantages.

As chips with larger number of components on them become more economic, more-sophisticated microcomputer architectures can be expected. For example, larger read/write memories will be available on the microprocessor chips themselves, and entire portions of CPUs will be duplicated or triplicated in order to simplify the handling of multiple processes or to provide self-checking.

Current microprocessor/microcomputer utilization is constrained by the past experience of users, system designers, and programmers, but the tremendous impact of this technology is producing a new breed of users. The microprocessor/microcomputer will soon be in the hands of a new generation of designers who are trained to regard it as a simple device, much as today's engineers regard the transistor and even the moderately complex integrated circuit.

C4.2.3.2 ARCHITECTURAL DESIGN CONCEPT. A baseline design concept for the 1985 NC^3N architecture is discussed in this section. Four design drivers are critical to the design of the NC^3N system:

- Modularity
- Automation
- Compatibility
- Cost

The requirements of technological adaptability and expandability are the principal factors that lead to the realization that functional and physical modularity is perhaps the single most important design driver that emerges from the system requirements. The reason that modularity is such a critical constraint in the system lies in the diversity of equipments and systems that are contained (or will be) within the variety of Naval platforms affected. The NC^3N can be applied in a cost-effective manner to all naval platforms only through modularity. It can be easily proved, on the basis of earlier system requirements studies, and from evaluation of other related shipboard systems, that without modularity, no one system design will satisfy the needs of all platforms. Therefore, this design criterion must pervade the design approach in all areas. While recognizing the importance of modularity, incorporating it in the baseline design entails a great deal of work; also, there remains the problem of achieving physical modularity in the implementation of these designs.

Aside from the advantages that accrue from new techniques and procedures, most of the advances in the NC^3N system must result from automation.

It must be the goal of the NC^3N automated design to reduce the required number of men to maintain normal operation of the system. This may be accomplished by designing the system to function entirely without human intervention when it is operating normally (without failure). This will relieve the operator of routine time-consuming tasks and allow him to spend his time managing the resources of the system.

To provide compatibility with other systems and equipment in today's platforms and the platforms of 1985, the NC^3N requires flexibility. This flexibility must be incorporated into both hardware and software designs on the basis of an architectural structure. The recommended design incorporates a modular software and hardware architecture that is capable of modification to accommodate equipment with changes only to the affected module, thereby minimizing the impact on the total system. This reduces debugging time when incorporating software changes and minimizes the risk of introducing errors in the remainder of the program. Hardware standardization of interfaces may be accomplished through the medium of standard types.

The last of the critical design drivers is cost – related to manning, training, logistics, and platform implementation. The most significant area of the design concept with respect to cost is processing. The architectural design concept of both processor hardware and software must be developed to achieve low cost while meeting the diverse requirements inherent in the variety of platforms the NC^3N must serve. The distributed processing approach provides cost benefits in both hardware and software. The hardware benefit accrues from the fact that use of the distributed processor element, implemented through microprocessor/microcomputer technology, allows the processing capability of the NC^3N to be closely tailored to the requirements of the particular platform. Since the distributed processing element is a low-cost item, the incremental step in cost associated with added processing is greatly reduced over that of minicomputers or centralized processing using large- or medium-scale computers. This results in an optimum relationship between processing capability and processing requirements. The attendant cost savings of this approach can easily be seen in any realistic cost analysis.

Experience indicates that software development costs for a system of this size and complexity will always at least equal and will usually exceed, by a factor of several hundred percent, the hardware development costs. The distributed processing architecture lends itself nicely to a modular software approach, which facilitates debugging – the major factor in software costs. Government surveys reveal that although the average programmer can generate several hundred lines of code per day, he can debug fewer than 10.

C4.2.3.3 SOFTWARE PHILOSOPHY. The NC^3N architectural concept evolved from the development of design drivers based on the technical and operational requirements. A distributed processing architecture based on microprocessor/microcomputer elements has been identified. The software (programming) concentrates on systems software architecture rather than detailed coding techniques. The software design methodology must be oriented toward the optimization of functional responsiveness, system modularity, formulation of precise documentation standards, structuring efficient software control, etc.

a. General Concepts

The software design structure recommended is organized on an upward-compatible, conformal set of modular building blocks. The modules will be controlled, during operation, by the NC^3N operating system through its executive, manager, and supervisory programs and will contain a high degree of commonality due to the similarity of tasks within different size ships. Different tasks (net controller vs net subscriber) will require a somewhat different mix of modules; but within any particular task the memory requirements will be a function of external stimuli, such as number of messages and number of controllable devices. The variables will not be imbedded in the building blocks, where they would act as system constraints.

During quiescence, the programs may be contained in a centralized program library applicable to all ships. Before the start of the mission, the programs appropriate to a specific ship's configuration may be selected from the library media (probably magnetic tape) and combined with the results of the data base generator (table generation, buffer sizing, etc) to become the "program load" selectively associated with each distributed processor in an explicit ship's configuration.

b. Introduction

Dispersion of C^2 tasks has been accelerated by miniaturization of computers. Considerations for the use of microprocessors in multiprocessor systems are similar to those

for the use of any other type of processor (minicomputer) with the exception of three characteristics – shared memory, low cost, and microprogramming capability. Most microprocessors have limited execution rates and this causes system throughput to be limited by the processor rather than memory. If this is still true in the 1980s, several microprocessors may be used in parallel and share the same memory without degrading individual execution rates. Their low cost (compared to the cost of memory and peripheral devices) allows the use of many microprocessors in one system. Microprogramming will allow the software instruction set to be tailored to equipment, functions, and control requirements.

Data acquisition, measurement and test, supervisory control, and computer communications are excellent candidates for use within hierarchical nodes. Microprocessors here function as subhierarchies of activity of varying size and functional capacity. They interconnect to a central control source (the CCU) which organizes the interconnectivity and control of the activity components which are themselves simultaneously active and physically and functionally distributed and have various separate real-time tasks.

The distributed processing concept discussed is asymmetrical, typically having dedicated applications wherein type, frequency of occurrence, and relative importance of task are known in advance. In this case processors may be specialized to carry out one particular type of task. For instance, one processor may perform all I/O operations, another provide encoding and decoding as well as implementing EDAC techniques. A third may provide configuration and reconfiguration control, etc. Specialization may occur via software programs to be executed, as well as the microprogram (which implements the processor instruction set) and hardware architectural features (number of registers, interrupt capability, stack processing).

Several advantages may be realized with multiprocessing systems in general. Throughput often increases almost directly with the number of microprocessors while system cost increases by only a small amount. Shared system resources offer an economic advantage by eliminating devices which would need to be duplicated in separate stand-alone systems.

With this mode of operation, homogeneous processes are clustered around a dedicated microprocessor, an Interface Central Unit (ICU), within the system. The ICU controls the processes and is responsible for the accuracy and correctness of the results accomplished by the particular process (interrogate monitor point, verify receiver configuration, switch equipment, etc). Depending on the variety of homogeneous processes, the ICUs could differ from one another in numerous ways. However, for commonality, the recommended approach allows variability only with regard to memory capacity, I/O channels, and number of microprocessors in a microcomputer. Common hardware modules are used throughout.

The decision as to which task should be assigned to which processor need not be made in real time by the system, simplifying software problems considerably. This burden may now be shifted to the systems designer, who must partition system requirements in such a way that each specialized processor is kept busy a sufficient amount of time to justify its existence. When low-cost microprocessors are used, utilization does not have to be high to justify the addition of a new processor. A side benefit of this partitioning is often simplification of programming, as each task can now be treated as an independent module with no provisions required for execution of other tasks by a given microprocessor.

In addition to being classified according to processor use, multiprocessor systems may also be grouped in relation to the interconnection of processors with systems memories and peripheral devices. The central control unit (CCU) coordinates activities at each ICU by requesting specific information and sanity checks and monitoring alarm

conditions. The CCU maintains records for display, logging, and reports. The CCU receives commands from the man-machine interface to order the ICUs to change limits or scanning variables. The CCU has therefore three principal tasks:

- Control. ICUs are interrogated on parametrically predetermined schedules to obtain preprocessed (by ICU) configuration status, monitor and test results, and alarm status or to issue configuration, frequency, and other change orders which are preplanned or dynamically entered by the operator.
- Processing. The data received from the ICUs are indexed for logging and prepared for summary reports. Secondary storage, in the form of magnetic tape and RO devices, provides the recording media.
- Man-machine interface. Interaction with the operator is monitored and controlled by the CCU. These facilities are used to support system direction, modification, and reporting.

The ICU, which is remoted from the CCU, but not necessarily distant from it, provides intelligence in three ways.

- Acquisition. The ICUs are organized homogeneously, each ICU being responsible for a set of function-oriented processes. By controlling the number and frequency of input variables, considerable versatility and parallel processing is provided. Slow- and fast-responding analog signals may be read at different time intervals; failure points may be selectively taken from scan, and multiple-thread reconfigurations implemented offset parallel (thread one starts at time T_0 and ends at T_9, thread two starts at T_1 and ends at T_{10}, times T_1 through T_9 occur in parallel).
- Processing. Collected data are converted into a structured file form suitable for presentation to the operator and storage in the history file.
- Display. The man-machine interface is simplified through program control. Displays are tailored to the process and are therefore easier to understand. The CCU controls the presentation to the operator of display types, such as switching and transmission, while the ICU controls the in-type display presentation. The operator, through the CCU, may select variables for display. Data or status conditions may be displayed selectively, cyclically, or in entirety. History of a process (static trace) may be obtained by logging stored data.

In order to take advantage of natural capability for graceful degradation and the inherent system availability associated with a distributed system, a strong operations monitor may be provided as the "watchdog." Each microprocessor will then contain a "process level" watchdog routine which is set to a time period determined by the longest acceptable microprocessor cycle. At the end of the cycle, the microprocessor supervisor resets the timer. Thus, if the cycle exceeds the predetermined worst-case maximum, the monitor will time out, since it has not been reset. At this point the watchdog monitor can interrupt the microprocessor, and the resulting interrupt service routine will alert the CCU (unless the failure is in the CCU) and notify the operator of the failure. If the failure is in the CCU itself, the arbiter will cause a switchover to the standby CCU, which will alert the operator and commence the recovery procedure. The technique has been used successfully in commercial applications. However, failure detected by this "functional level" watchdog monitor indicates microprocessor failure. At the system level, the CCU is exchanging sanity check messages with all ICUs. Lack of receipt of a system level sanity message alerts the ICUs that the CCU is in transition from standby to prime, thereby forcing a status monitor

queue buildup until the CCU has fully recovered and sends the required sanity message. The CCU system level watchdog routine is continually monitoring sanity messages being sent from the ICU system level watchdog. If this sanity message fails to appear, the operator is notified of a malfunction in the negligent ICU. If the message is sent garbled, the operator is notified of a malfunction in the ICU I/O or memory. Status information of this type (such as process level, function level, and system level watchdog results) is reported to the operator for his initiation of diagnostics and other measures to ensure process security and accuracy.

The watchdog routine's sanity messages must contain intrinsic error control such as horizontal parity, echo checking, vertical parity, or hash totaling. It is of the utmost importance for the system to ensure against the possibility that functioning equipment is stated to be malfunctioning by a malfunctioning control technique. This is best accomplished by distributed processing, since the intranodal checks and balances associated with the system galaxy concept contain inherent "communal health" characteristics. Remote computer control via a large centralized processor and local operator control provided through hardwired logic are the methods presently used to accomplish the functions discussed here. Modularity introduced by distributed microprocessor control offers the following significant advantages over those alternatives:

- Software (or firmware) at each location may be tailored to specific I/O configuration and functional requirements at the site. This eliminates complicated and costly software operating systems ordinarily required for a large central computer to process different signals at various locations. If a remote site configuration does change, only the local microprocessor is affected. In a remote control environment, the entire system would have to be taken off line and tested for each remote site change. In addition, microprocessor controlled remotes may be added or removed without disturbing the rest of the system.

- Partitioning of system functions between master and remotes translates into a lower speed and processing power requirement at the master, even under high volume, and, hence, lower cost. Parallel operations such as data acquisition, digital filtering, and formatting are performed at the remotes while the master manages and analyzes data flow.

- The designer may use the same component at each site and simply change the program (or ROM). Such standardization reduces design time, enables the system to adapt easily to changing requirements, and requires less documentation, training and maintenance. This can be favorably contrasted with the multiplicity of medium-scale integration (MSI) components required for a hardwired system.

- Local processors remain operational if the master is down, assuring maximum uptime for real-time applications. Collected data are processed and saved in local memory. This reliability improvement is made possible only through intelligent local control.

- By using microprocessors rather than minicomputers or random logic, storage area, power consumption, and cooling requirements are reduced.

c. Software Architecture

The operational software system is structured in two generic subsystems:

The operating subsystem

The applications subsystem

The subsystems are segmented into a number of programs, each program responsible for the processing of a particular general function within the generic subsystem. Each program is then broken into tasks which are accomplished by routines, if unique to the program, or

subroutines, if the task can be generalized across program boundaries. Since the environment is monoprogramming, "task" refers simply to a single activity having clearly boundable constraints. A "routine" is a set of computer instructions and associated data constructed so as to accomplish a specific function. There are three distinct hardware/software relationships which must be considered:

Internodal

Intranodal

Man-machine

The internodal relationship is accomplished by the external interfaces or link adaptation functional area or group of microprocessors, as it pertains to traffic; and a combination of the Management and Control (M&C) functional area or group of microprocessors with the Network Access Switching functional area or group as it pertains to equipment configuration. The intranodal relationship, which is basically involved in nodal supervision, health, and control, is accomplished by the operating system (OS) components. The man-machine relationship is handled by the MCG M&C group and the operator using the various peripherals as media.

1. Internodal Concepts

The link adaptation function is divided into two parts:

Transmission control

Traffic control

The latter has a very important adjunct – packet management. Transmission control is responsible for I/O interfaces for all types of transmissions -- point to point, network subscriber/controller, link quality, and broadcast. This program will pack each eight bits received from the I/O interface into a homogeneous buffer which, when it reaches a parametrically prescribed limit, will alert the traffic control program to the location of this buffer of unprocessed data. This program will, in certain cases, determine whether the message is for this platform because of addressing or because the ship's captain wishes to receive the information for monitoring purposes. The traffic control program will be alerted through the message index, which contains the necessary characteristics and supporting data concerning the unprocessed data. The traffic control program will accumulate these buffers and commence to "thread" them into messages. Once the message has been constructed, the traffic control program will begin to process the data, all EDAC and data manipulation will be accomplished and the message will be readied for packet management or transfer to information processing, if this is input; or transmission through the TX, if this is output. Therefore, an adjunct to the traffic control program is a packet/IP manager program which functions as the channel continuity control between the IP and the NC^3N; and, if required, provides packet management, such as packetizing on input from IP, depacketizing on output to IP, for the system.

The Internodal Equipment Configuration is concerned with configuration and/or reconfiguration of switchable devices which may be associated in a manner allowing for receipt or transmission of data. Therefore, the M&C group reviews system status, determines equipment availability, and issues commands to the particular ICUs which cause the chaining of the necessary equipment. Once the chaining has been accomplished (the desired thread is tested first by an end-to-end test message, if possible; if not, by a closed-loop test message), the configuration status table is updated and the change in configuration is reported to the operator.

2. Intranodal Concept

Intranodal system management and control data are in support of the maintenance philosophy and the man-machine interface. Three forms of system assurance occur:

Sanity checking

Link quality monitoring

Equipment diagnostics

Intranodal communication provides the medium for "watchdog" transfers, link quality monitoring messages, and directives for equipment diagnostics.

All three forms are used to improve system reliability and availability. The equipment diagnostics philosophy subdivides into fault detection, an on-line monitoring function; and fault isolation, an off-line test function. The fault detection monitor points are intrinsic to the real-time processing and are performed in the ICUs responsible for the equipment subsystem being monitored. This monitoring is done by CCU direction, if unscheduled; and automatically, if scheduled. The testing is accomplished off line at the operator's convenience. The off-line test is used both to isolate a fault to a replaceable module and as an assurance test on the replacing or "fixed" module. This type of activity necessarily requires operator intervention in order to decide when to run the test and whether sufficient equipment is available to run it; to set up the test loop; and to supply the test input which will provide the necessary certitude to the test. This leads directly to the third internodal relationship, the one between the system and the operator.

3. The Man-Machine Relationship

One key ingredient is finally needed – someone rational, who can make decisions based on a multitude of inputs received simultaneously or on a repeat of a historical pattern. This is the operator; his medium for directing the system to perform a series of functions is the intranodal communication and his message (means) is by action entry, using as a mode some form of keyboard console or voice. The action entry may be complex, as in the case of a failure which required a reload of a microprocessor from a magnetic tape segment; or simple as in changing a monitor point delimiter in the monitor table of acceptable ranges. The action entry itself will be of a predetermined form allowing the operator as much latitude for construction as possible.

4. Programming Standards

In order to improve clarity, ensure against self-definition, assist in life-cycle maintainability, and provide a coherent presentation, definite consistent rules for coding and annotating program listings must be followed.

5. Utility Programs

A set of utility routines must be developed. Most will be stand-alone (not under OS) and must be individually loaded by the operator. Obvious inclusions are:

Assembler/Compiler	(modular/stand-alone)
Data Base Generator	(modular/stand-alone)
Test Data Generator	(modular/stand-alone)
System Bootstrap	(modular/stand-alone)
System Loader	(modular/stand-alone)

System Recovery	(modular/stand-alone)
Memory Dump	(modular/stand-alone)
Tape Dump	(modular/stand-alone)
Tape Labeling	(modular/stand-alone)
Memory Trace	(under OS)
Program Trace	(under OS)
Inspect and Change	(under OS)
I/O Handler	(both)

(limited to nonoperational modes)

C4.2.3.4 NC^3N SYSTEM HARDWARE. The basic hardware building block of the NC^3N architecture will be the microcomputer, which will be referred to as a distributed element (DE).

The DE will function separately to act as Standard Interface Units (SIUs); and in distributed functional groupings to provide management and control, cryptography, switching, etc. In addition, the DE is used to implement the LAG function by distributing the processing among several DEs. Hence, a common DE design can be used for all NC^3N functions.

The DE concept has led to the consideration of distributed processing implementation for all the functional area requirements. In distributed processing, the processing load is divided among several processing units, the number relating to the size or complexity of the functional tasks and also to platform type and mission requirements.

In the distributed processing implementation, the processing and control associated with a system is accomplished by an interface control unit (ICU) dedicated to that system. The ICU is responsible for all control, monitor, and test functions of its dedicated equipment. Each ICU has an I/O interface which allows it to interface with the equipment for which it is responsible. Most M&C activities require action and coordination by several systems. To configure a link, for example, requires action by some or all of the following systems: transmission, man/machine couplers, information processing. The question arises as to how to coordinate the different system activities. Two basic approaches are possible. The first requires an interconnection between all the ICUs. The second is a central processing concept which dictates the need for a central processing unit whose function is to manage and coordinate all the ICUs dedicated to system control. The latter approach minimizes the interfacing among the ICUs and lends itself to optimum functional distribution among the ICUs. This central unit is referred to as the CCU. The CCU is responsible for coordinating the system ICUs. In addition, it is also responsible for gathering key parameters from the ICUs in order to perform the M&C coordination, decision making, display/alarm, record keeping, and reporting functions.

The distributed processing concept has been made feasible by the advent of the microprocessor. Prior to the development of the microprocessor, this concept would have required that a minicomputer be dedicated to each system. Hence, this would have required a multitude of minicomputers instead of one or two central minicomputers, as in the case of the central processing implementation. The miniprocessor cost and complexity formerly made the consideration of this concept impractical. The availability of the microprocessor, however, makes this implementation cost-effective.

The basic building block for distributed processing being considered here is the distributed element. The DE processing capabilities are handled by one or more microprocessors within the DE. In addition, each DE contains an I/O interface which allows it to interface with the system equipment. Each ICU consists of one or more DEs: the number of DEs within each ICU is a function of the number of equipments dedicated to the ICU and the complexity of the equipment being controlled. Each DE has a standard set of interfaces: parallel channels, serial channels, dedicated analog lines, and dedicated digital lines. Hence, all DEs within the ICUs are identical, differing only in the number of the standard interfaces employed and in the amount of memory, which is a function of ICU processing requirements. The CCU consists of two DEs. One DE is at all times in control of the M&C operations. The second is a backup to the first in case of failure.

It is of paramount importance to have a set of common hardware used throughout the NC^3N. There are several major advantages to this approach:

- Smaller inventory of spare parts.
- Ease of maintenance. Service personnel have to be trained to service only one common set of hardware. In addition, an exhaustive set of fault detection and fault isolation procedures can be developed which can be adapted to each major system component.
- Development and design costs are reduced.
- Lower production costs.

The distributed processing implementation is accomplished by employing the distributed elements. All DEs have the same processing unit. Each DE memory consists of scratch pad and program memory. All scratch pad memories (typically RAM) will consist of the same devices, the only difference being the amount of scratch pad memory required by each DE. This poses no problem since this part of memory can be modularized so that the only difference between DE scratch pad memories is the number of modules. Similarly, all program memories will employ the same devices (typically RAM, ROM, or PROM). If it is shown to be advantageous to use RAM for program memory, then the scratch pad and program memory, then the scratch pad and program memory consist of the same modules. Hence, in this case the only difference between the DE memories is the number of modules required. The ICU distributed element I/O interface consists of controller-to-controller interfaces for ICU to CCU communication and controller-to-system interfaces for communication with the equipment. All ICUs use the same set of standard controller-to-system interfaces. The CCU distributed element requires controller-to-controller interface for ICU to CCU communication. In addition, it requires I/O interfaces for the peripheral equipment. This interface can use one of the standard interfaces provided for ICU to system communication. Therefore, the CCU employs the same standard interfaces as the ICU. Hence, all distributed elements are identical, differing only in the number of memory modules and the number of each of the standard interfaces employed. From this, it can be seen that the distributed implementation allows a high degree of common hardware within the NC^3N.

One of the major design goals of the NC^3N is to use a small number of simple building blocks to implement the system functions. Besides the reduction in development, design, fabrication, and integration efforts, the benefits of this approach can be extended all the way to the maintenance level. In addition, the ability to adapt the NC^3N to different classes of ships and new electronic technology provides extreme simplicity and adaptability.

The simplicity of programming architecture is also an important consideration. Each distributed element performs relatively few functions; hence, the effort is reduced to

integrating a number of programs, one for each distributed element, each consisting of a reduced number of molecules. Programming integration and debugging are simplified under the distributed processing approach.

The distributed processing implementation discussed here uses a central control element which coordinates the interface control units. If redundancy were not a consideration, the central control unit would consist of a single distributed element. However, in case of failure of the central control unit distributed element, there is a loss of system control. The simplest solution to this problem is to include two processing elements within the central control unit. Due to greatly reduced cost and complexity of the distributed element system, degradation due to failure by a distributed element within the CCU can be prevented with little overall cost and effort. Failure of a distributed element ICU is far less catastrophic, because only automatic control and monitor of the equipment dedicated to the failed unit are lost.

One of the major NC^3N requirements is that it must be adaptable to different classes of ships and to a variety of transitional equipment within a ship as current manual or semi-automatic equipment is replaced by new standard automatic equipment. Therefore, adaptability is a must for NC^3N design.

As the complexity of the node is reduced, the NC^3N must be capable of being reduced proportionately. The other aspect of adaptability which must be considered is the NC^3N ability to handle complex node requirements. As system complexity increases, the computational capability of the system must be able to accommodate the increased demands placed upon it, until its capabilities are exceeded.

For architectural simplicity, it is desirable to have the ICU perform the functions associated with its equipment control, while the central processing unit coordinates all the ICU activities. When the computational capabilities of an ICU or central control unit are exceeded, another distributed element is added to the ICU or central control unit. Because of the low cost of a distributed element, this has little effect on system cost. By adding another distributed element, the functional split among the ICUs and central control unit is maintained. The second approach is to utilize the optimum loading capabilities of distributed element increase, the functions can be redistributed among the distributed elements. If, for example, the processing burden on an ICU increases, that ICU still performs all the interfacing with its dedicated system; however, some of the processing is performed either in the central control unit or in the other ICUs. Hence, if the capabilities of a distributed element are exceeded, this problem can be solved without adding another distributed element to the system by sacrificing the functional split between the ICUs and the central control unit.

The SIU concept requires that all the NC^3N equipment interfaces conform to one standard interface design. This can be readily incorporated into the distributed element interfacing with the equipment. While the I/O channel hardware interfacing directly with the device must be modified, a great deal of interface modification can be accommodated through the distributed element software. The high degree of commonality between distributed elements is still achieved.

C4.2.3.5 INTERIM CAPABILITY. There is a need to begin introduction of NC^3N capability prior to the 1985 time frame so that existing C^2 deficiencies may be eased in the meantime. This requires a plan to provide a bridging or interim NC^3N. This brings us to perhaps the most important ramification of modularity, which is evolution.

There will be many phases of evolution between now and 1985. From 1985 on, the process will take smaller technological steps and will be less difficult. The distinctive feature about the period from now to 1985 is seen by considering the terminal dates. For example, most shipboard radio equipment now in use is manual, whereas in 1985 the corner will have been turned and most radio equipment will be automated. During that period, then, drastic changes will take place with respect to the performance requirements imposed on the NC^3N. These changes must be adapted to go through the modularity of the NC^3N, special interfaces, and modification of radio equipment as determined by cost-effectiveness.

The evolutionary capability of the NC^3N is of paramount importance to:

Life-cycle cost-effectiveness

Technological adaptability

Expandability

The NC^3N will be an evolutionary design by virtue of carefully designed modularity through the architectural concept of microprocessor/microcomputer distributed processing. Technological adaptability is the capability to accept new systems with minimum impact. To illustrate, let us assume that in the post-1985 era an optical communications link is to be introduced into the fleet. This system may have a 9600-baud interface at baseband with 50 baseband channels and a control interfaced with the MCG. To accept this into the NC^3N will be a matter of adding (as required) switching modules to accommodate the increased channels, an I/O card in the link-ICU for control and monitor, and modified ROMs in the link-ICU and CCU. The cost is minimized by the absence of non-communication-oriented requirements in the optical system itself and in the NC^3N. No special display or control panel is required, no additional control cabling is required, personnel training is restricted to the radio (optical) portion of the system, and other systems need not be displaced until the new system is proved.

Expandability is the ability of the NC^3N to accept more or less equipment of the variety in use at a fixed point in time with minimum impact. For example, suppose a guided missile cruiser (CG) has been serving in a large task force as a member of an escort unit for a long time and then, after periodic maintenance overhaul, is to be assigned as unit command ship in an open ocean sea control group. Then, during the overhaul the CG will be outfitted with the required radio equipment and modules. With each additional radio equipment there will be associated a set of software and hardware modules for control, monitor, test, link adaptation, and switching, all of which will be common to those already existing in the CG. This approach will expedite the incorporation of new equipment into platforms and provide attendant cost savings. Changes to the system to accommodate additional equipment in other nodal systems amount to increasing the number of modules rather than adding new equipment.

Interim capability will be basically achieved through the inherent modularity of the NC^3N. Modularity will be achieved through the use of basic modular elements of hardware (microprocessors/microcomputer) and software in a distributed processing framework.

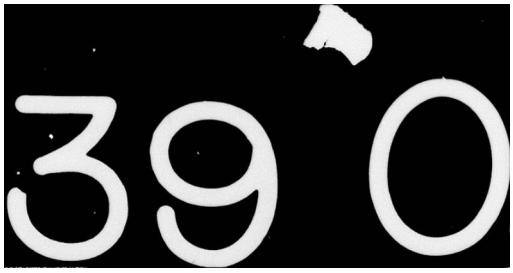

EN

AD-A039 026 NAVAL ELECTRONICS LAB CENTER SAN DIEGO CALIF F/G 17/2
NAVY COMMAND CONTROL AND COMMUNICATIONS SYSTEM DESIGN PRINCIPLE--ETC(U)
AUG 76

UNCLASSIFIED NELC/TD-504-VOL-4 NL

2 OF 2
AD A039 026

END
DATE FILMED
10-77
DDC


390

SUPPLEMENTARY

INFORMATION

NAVAL OCEAN SYSTEMS CENTER
San Diego, California 92152

1 July 1977

LITERATURE CHANGE

NELC Technical Document 504
NAVY COMMAND CONTROL AND COMMUNICATIONS SYSTEM DESIGN PRINCIPLES AND CONCEPTS, Volumes I through VIII, 15 August 1976

1. In block 10 of DD Form 1473, change numbers to:
   65866N, X0740, X0740 (NELC Q221)

2. On cover, under date, add:
   Changed 1 July 1977